Call-A.P.P.L.E.

July 1984
Volume 7, Number 7

Worlds Largest Apple User Group

$3.00
$3.75 Canada
£2.00 UK

COMPUTER LANGUAGES

Comparison of Two Logos

Bringing Up FORTH

Hilbert and the Alligators
(A Graphic Benchmark)

Managing Ampermanager

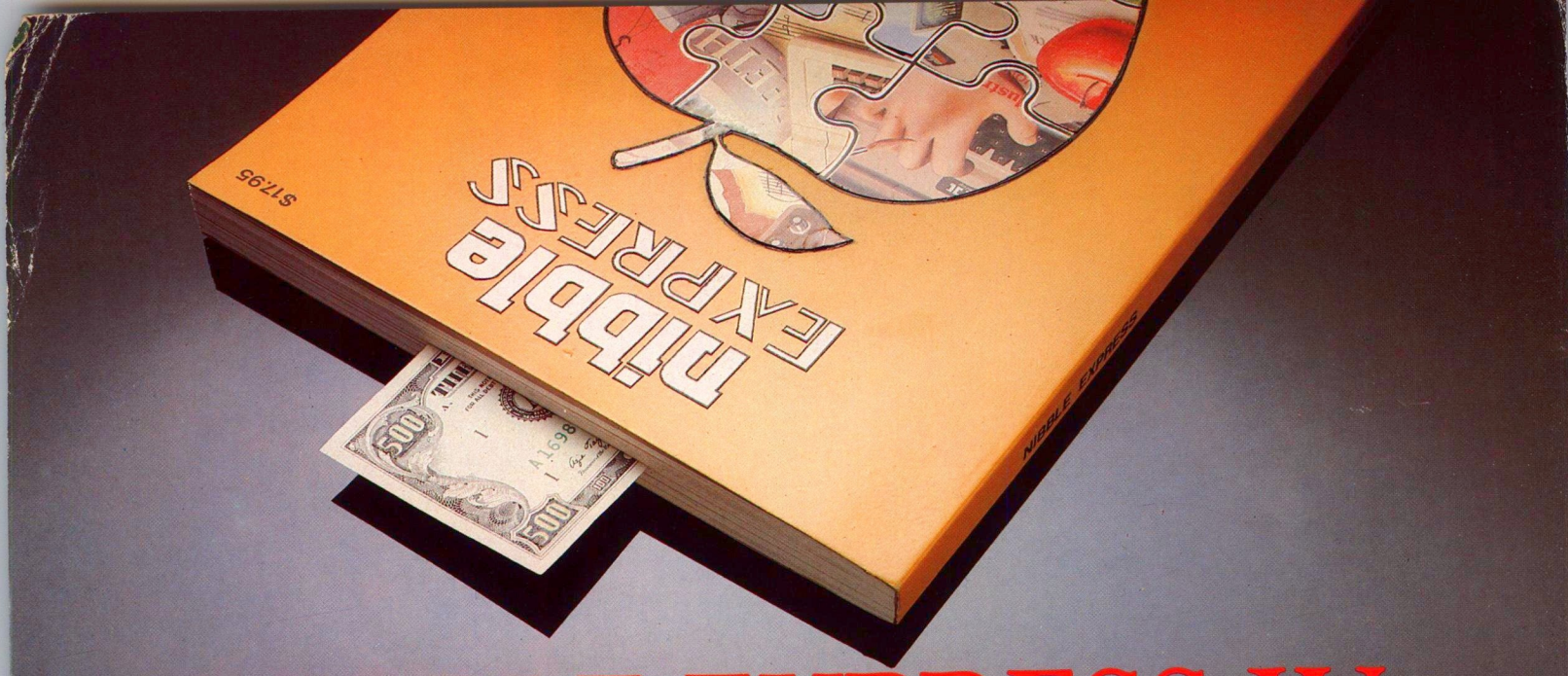
Running Applesoft with ProDOS

B. Kelsten

REVIEWS:
— Microsoft BASIC on the Macintosh
— Foreign Language Learning Software
— P-LISP

INSIDE: Up-to-the-minute information about Macintosh, ProDOS, and more. . .

Apple PugetSound Program Library Exchange



NIBBLE EXPRESS IV

The $500 Program Book for your Apple

That's right! More than $500 worth of programs for your Home, Business, Education and Entertainment. NIBBLE EXPRESS IV is an Anthology of the Major Articles and Programs appearing in Volume 4 (1983) of NIBBLE Magazine.

NIBBLE EXPRESS IV contains *up-to-date enhancements* and *complete program listings* with comprehensive *articles* that show:

What each program does.

How to type them into your Apple, ACE, or other Applesoft-compatible computer.

How to use them.

How to customize them to your own needs.

Here's a representative sample of the 50 programs and articles:

The Investor—*Stock Tracking, Reporting, and Graphics.*
The Librarian—*Automatic control of your Disk Libraries.*
The Designer/Illustrator—*Art Design/Creation and Composition.*
RAM Manager—*Tuck away up to 10 Programs in RAM Card Memory.*
The Programmer—*Applesoft Preprocessor for Structured BASIC.*
T.U.N.E.S.—*Music Composition and Playback.*
RAM Disk—*Make a Phantom Disk out of your 16K RAM Card.*
The Nibbler—*Arcade Action with the famous Pellet Eater.*
Math Monster—*Arithmetic can be fun.*
Personal Appointment Calendar—*Track events, dates, and meetings.*

AND THERE'S MORE...

The NIBBLE EXPRESS is full of useful Tips 'N Techniques, Utilities, and other Goodies for getting more out of your Apple.

Even if you own all 8 issues of NIBBLE in 1983, you'll want the EXPRESS IV to have updated program listings in one convenient package. It's a MUST for your library.

nibble P.O. Box 325, Lincoln, MA 01773

Yes I want NIBBLE EXPRESS in my library!

Send me:

- ☐ Nibble Express Volume I at $14.95 each.
- ☐ Nibble Express Volume II at $14.95 each.
- ☐ Nibble Express Volume III at $17.95 each.
- ☐ Nibble Express Volume IV at $17.95 each.

Please add $1.75 shipping/handling per copy. (Outside U.S. add: $2.75 surface or $5.00 Air Mail per copy)

Enclosed is my: ☐ Check ☐ Money Order

Bill my: ☐ MasterCard ☐ VISA

Card # _____

Expires _____ Phone _____

Signature _____

Name _____

Street _____

City _____

State _____ Zip _____

Your check or money order must accompany your order to qualify. Outside U.S.: Checks must be drawn on a U.S. bank. *Apple is a registered trademark of Apple Computer Company.

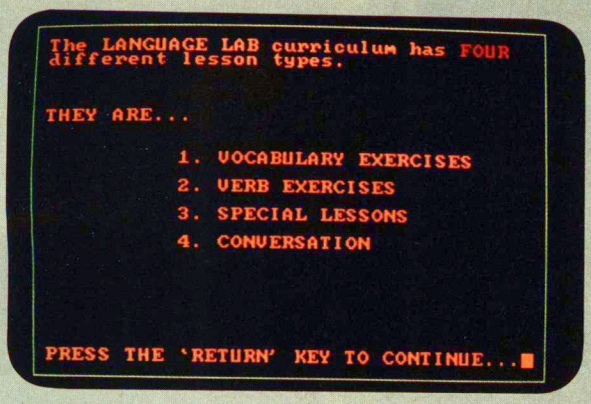
Circle 63 on Reader Service Card

Language Lab,™ The Only Complete Computerized Foreign Language Learning System

Learn a foreign language using your Apple or IBM personal computer.

All Needed Language Learning Skills...

A number of skills are required for fluency in a foreign language. Language Lab offers practice in all them.


The LANGUAGE LAB curriculum has FOUR different lesson types.
THEY ARE...
1. VOCABULARY EXERCISES
2. VERB EXERCISES
3. SPECIAL LESSONS
4. CONVERSATION
PRESS THE 'RETURN' KEY TO CONTINUE...

Interactive Audio Cassette

Proper pronunciation is perfectly coordinated with lesson material presented on your video display. Both audio and video presentations compliment each other at exactly the right moment.

Add-On Languages and Learning Levels

Language Lab allows you to add additional foreign languages and proficiency levels to your library of lessons. Once you have purchased Language Lab, you may for a nominal charge utilize Language Lab for the study of any Lesson Level or Language offered.

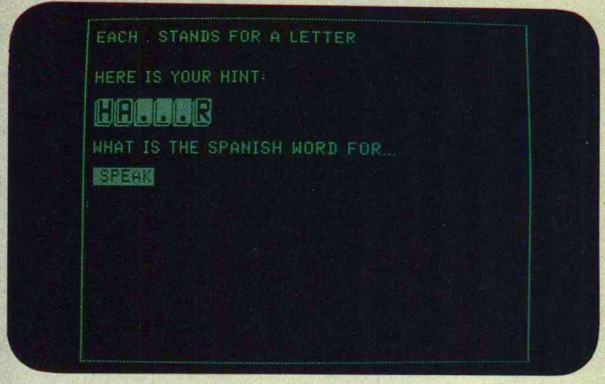
Custom Curriculum

Custom Curriculum allows you to modify the contents of the lessons presented by Language Lab, or create entirely new lessons. This program is ideal for Teachers and Students alike, who desire to make lessons consistent with language lesson material in use in the classroom.

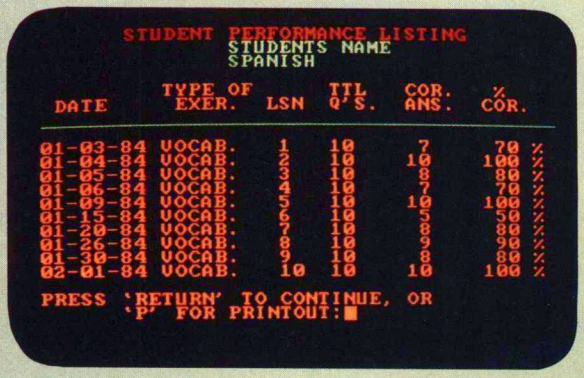
A variety of displays reinforce each exercise, creating greater lesson retention with exciting and stimulating computer generated graphics.

Hints and Clues is one of the many programs that helps you learn while working with Language Lab.

With this unique approach to self testing, the Language Lab System increases your retention and ability to utilize your newly acquired language.



Check on your progress as you proceed with your study. Language Lab shows your strengths as well as your weaknesses.


STUDENT PERFORMANCE LISTING
STUDENTS NAME
SPANISH
DATE TYPE OF EXER. LSN TTL Q'S. COR. ANS. % COR.
01-03-84 VOCAB. 1 10 7 70 %
01-04-84 VOCAB. 2 10 10 100 %
01-05-84 VOCAB. 3 10 8 80 %
01-06-84 VOCAB. 4 10 7 70 %
01-09-84 VOCAB. 5 10 10 100 %
01-15-84 VOCAB. 6 10 5 50 %
01-20-84 VOCAB. 7 10 8 80 %
01-26-84 VOCAB. 8 10 9 90 %
01-30-84 VOCAB. 9 10 8 80 %
02-01-84 VOCAB. 10 10 10 100 %
PRESS 'RETURN' TO CONTINUE, OR 'P' FOR PRINTOUT:

LampLighter Software, Inc.
7 Breton Avenue
Melville, New York 11747

(516) 421-2653

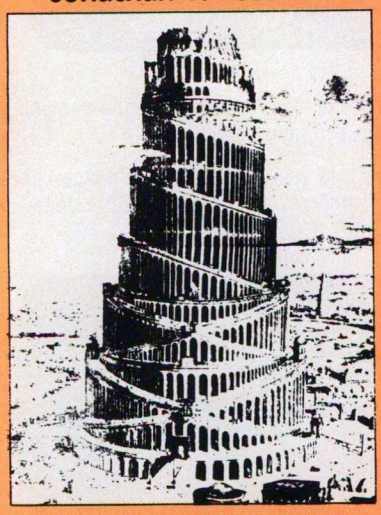
Circle 67 on Reader Service Card

Official Publication of the World's Largest Apple User Group

Call-

Language Features

6 Computer Languages
Jonathan V. Post



Are we building a high-tech Tower of Babel?

8 Bringing up FORTH
Andrew Bernat

Implementing this special language on the Apple II

18 Hilbert and the Alligators
Don Williams

A graphic benchmark for comparing several languages and systems

Random Access

30 Northern Spy: Managing With Ampermanager
Rick Sutcliffe

All about a useful, inexpensive utility, including a hybrid DOS 3.3 / ProDOS disk

41 Something Over 32 Bits
Dave Redhed

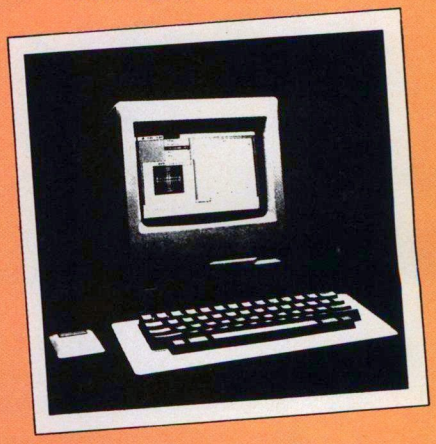
A new question/answer column about Apple's Macintosh and Lisa computers

43 Running APPLESOFT Programs With ProDOS
Richard D. Norling

Making the best use of ProDOS with your BASIC programs

Reviews

25 Microsoft BASIC on the Macintosh
Tom Sethre



Is this what Mac users have been waiting for?

28 LISP on the Apple II
Vince Giambalvo

A microcomputer version of the list-processing language

40 Book Review: Apple Assembly Language
Lee Webster

This book might help you learn it yourself

ADVERTISER INDEX

A.P.P.L.E. 37
Access Micro 57
ActiVision 59
Baudville 5
Beagle Bros. ..... 38 & 39
Bytes & Pieces 49
ComputerEase 22
David Data 11
Digital Vision 31
Encore Ribbon Services 42
Haba Systems C3
Hardcore Computist 21
Hollywood Hardware 29
Koala Technologies C4
Laing Electronics ..... 50 & 60
LampLighter 1
MagicSoft 47
Micron Technology 27
Name • Brand Software 55
Nibble Express C2
Nibble Notch 52
Quorum International 23
Schwartz Computer Consulting 63
Soflight Software 15
Softronics 64
Southeastern Software 19
Springboard Software ..... 32 & 33
StarLogic 9
Street Electronics Corp. 7
Technology One 61
Telephone Software Connection 53
The Professor 17
Trac Systems 35
V.F. Associates 62
Volition Systems ..... 44 & 45

A.P.P.L.E.™

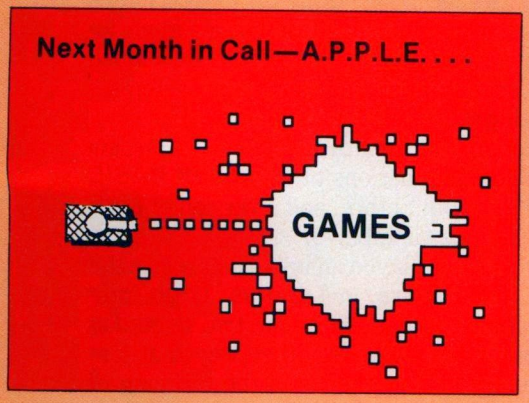
Apple PugetSound Program Library Exchange

Vol. 7, No. 7 July, 1984

Education

14 Speaking of Tongues: A Review of Foreign Language Learning Software
Kaiulani Schuler
Evaluating a wide range of language teaching products for the Apple

51 Logo vs. Logo
Jeff Sandys
A detailed comparison of the MIT and LCSI versions of Logo

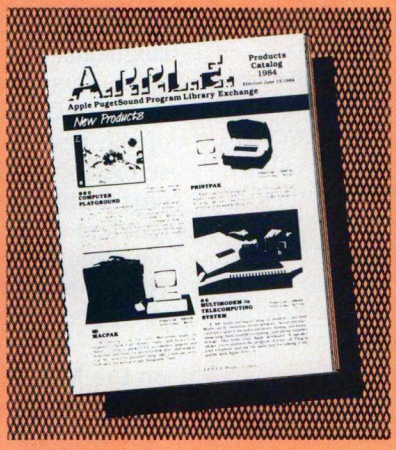


Departments

4 The INNER CIRCUIT
Don Elman
Can we sort out the babble of computer languages?

35 A.P.P.L.E. Member Services

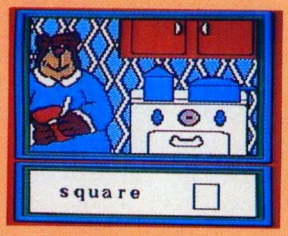
36 Member Billboard
Don Elman



Two announcements: release of the best catalog ever, and installation of a new electronic bulletin board system

37 New A.P.P.L.E. Products
The latest additions to our catalog items

58 The Marketplace



News releases and product information relating to languages, Macintosh, Apple IIc, and more

63 Unclassifieds

Letters

54 A.P.P.L.E. Doctor
Bruce F. Field
Prescriptions for your electronic aches and pains

56 Write—A.P.P.L.E.
Readers' forum

Apple user groups may reprint without prior permission any portion of the contents herein, provided author, title, and publication credits are given and a written notice of such use is furnished to the editorial office. For commercial applications of programs herein, the author should be contacted c/o General Office. Call — A.P.P.L.E. is published monthly by Apple PugetSound Program Library Exchange, 21246 – 68th Ave. S., Kent, WA 98032. Copies are mailed free as a benefit of A.P.P.L.E. membership. For copies received outside the U.S., please refer to Foreign Postage Chart on Order Form. Subscriptions alone are available at $21 per year and do not include membership.

Entire contents copyright © 1984 by Apple PugetSound Program Library Exchange.

Moving?

Changes of address must be received at the General Office at least 30 days prior to effective date of move. Issues missed due to non-receipt of change of address notification must be separately reordered by the member at current cost.

Please fill out and return this notice six weeks before moving.

PRESENT:	***NEW ADDRESS:***
NAME _____	*NAME* _____
ADDRESS _____	*ADDRESS* _____
CITY _____	*CITY* _____
STATE _____ *ZIP CODE* _____	*STATE* _____ *ZIP CODE* _____
EFFECTIVE DATE _____	*MEMBERSHIP NUMBER* _____

Call-A.P.P.L.E.

Gary Kruger	Publisher
Donald Elman	Editorial Coordinator
Kaiulani Schuler	Editorial Operations
Ed Adams	Art Director
Mary Hubert	Production
Ron Keister	Cover Artist
John Elliott	National Display
(206) 872-2245	Advertising

The INNER CIRCUIT

Don Elman

Skyscraper of Babel

IN ONE OF the early scenes of *Star Wars*, Luke Skywalker's Uncle asked C3PO, "Do you speak BACCI?" "Certainly, sir," came the 'droid's reply, "it's my native language!" That robot's prime function, in fact, was to be a translator among the myriad forms of living and electronic communication in that mythical galaxy.

Do we need a C3PO on twentieth century earth? Since at least Biblical times, the multiplicity of human languages and dialects has confounded the exchange of information between people. Only in recent decades have the jet plane and telecommunication satellites begun to set the stage for even a small degree of universality in spoken and written language.

At the same time, the development of computers has created a new dimension of informational confusion that was unimagined in the days of Babel. Not only do we still have to grapple with the problem of getting ideas from one human brain to another, but now we must attempt to transport thoughts between the unfathomable complexity of biological synapses and the structured rigidity of electronic microcircuits. Since no one has yet come up with a perfect medium for this type of unnatural communication, numerous functional (but imperfect) systems have been devised.

Who's In Charge?

At the heart of the computer language issue is the question of control. After all, the whole point of having such languages is to give the human user a means of directing what the computer does. One way to distinguish among various computer languages is the degree to which the person actually controls the computer—or vice versa. The original computer programmers had no choice but to put their commands in the on-off, binary form that has meaning to the CPU's innermost circuitry. Even today, many programmers pride themselves on the ability to communicate at or near "the machine's own level." Although this approach does afford nearly complete control over the individual bits within the computer, one drawback is that the programmer's entire thinking process must be reshaped—at considerable effort and frustration—into a form dictated by the computer's unique architecture. Once this transformation is accomplished, it is no longer clear exactly who is the master and who is the servant.

Using higher level languages both strengthens and weakens the programmer's degree of mastery over the computer's activities. The closer a computer language approximates the user's native tongue, the less he must bend to the computer's idiosyncratic structure in order to get something accomplished. On the other hand, the multiple translation steps incorporated into high-level languages create functional and syntactic limitations on what the programmer can make the computer do. Furthermore, the overwhelming variety of high-level languages today—each with its ardent band of supporters—is enough to make the most hardcore computerphile weep.

What More Can Be Said?

In this issue, we'll look at a handful of the many languages and language-related applications for Apple computers. Jonathan Post provides a keynote for our theme in **Computer Languages**. How to implement one of the newer generation languages on the Apple is described by Andrew Bernat in **Bringing Up FORTH**. The next feature article has a whimsical title but a serious intent and shows how a unique benchmark was developed for comparing the speed of different language systems—**Hilbert and the Alligators** by Don Williams.

Our review section this month also focuses on languages, with detailed examinations of **Microsoft BASIC on the Macintosh** by Tom Sethre, and a version of **LISP on the Apple II** by Vince Giambalvo.

Languages have always played an important role in the field of education. Jeff Sandys, in **Logo vs. Logo**, presents a point-by-point comparison of the two main dialects of this computer language. Another way that languages and computers go together is in the teaching of foreign languages with computer-assisted instruction. **Speaking of Tongues**, by Kaiulani Schuler, reviews a wide range of the language learning software that is available for the Apple.

This month we are pleased to introduce a new section focusing on Apple's 32-bit technology (Lisa and Macintosh) with **Something Over 32 Bits** by Dave Redhed. Also, of course, you'll find our usual variety of valuable technical information on such topics as amper utilities and ProDOS.

Climbing the Tower

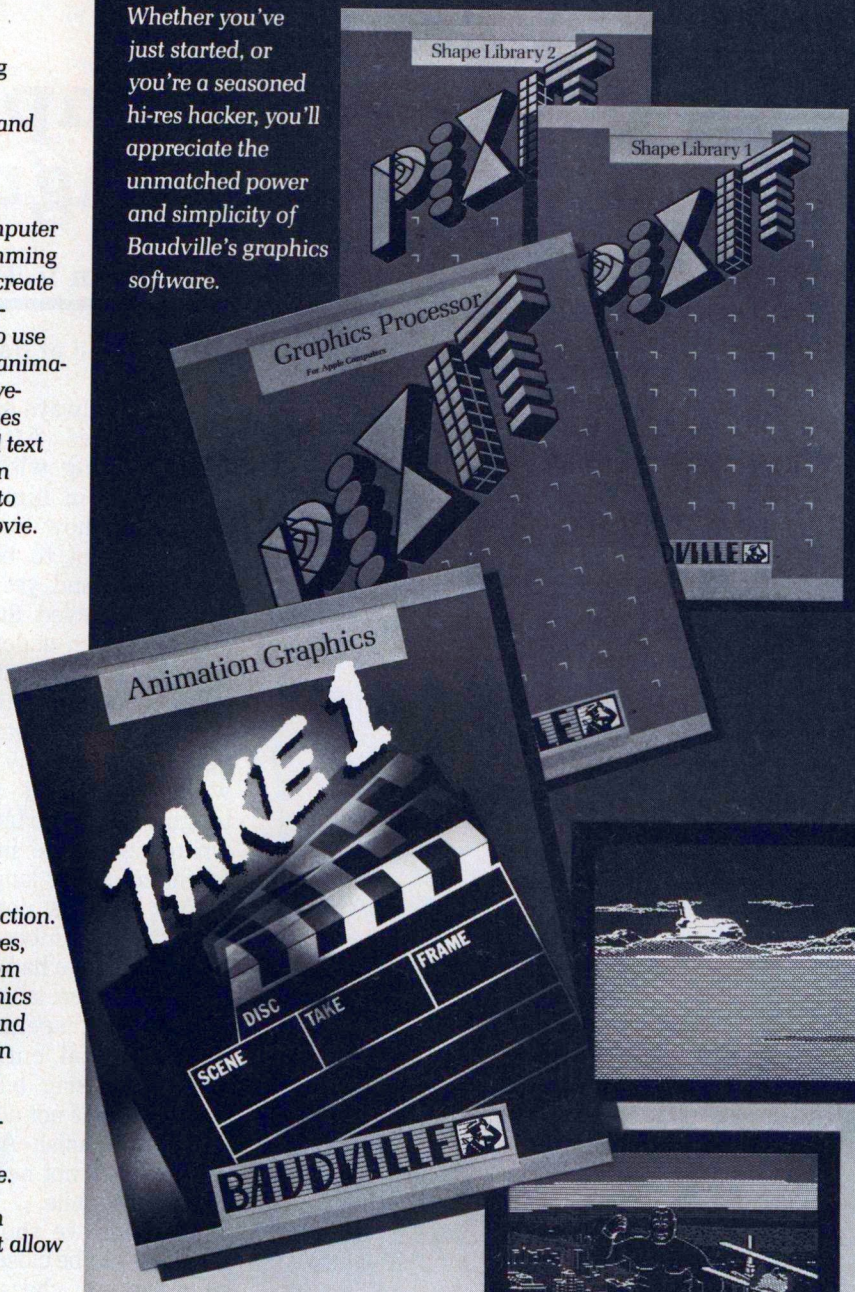
Like the residents of Babel, our attempts to find ultimate truths may be thwarted by a confusion of computerese tongues—but that probably won't dissuade us from the search. Now, in the meantime, if only I knew where to send away for a C3PO kit...

Picture it!

Graphics processing that's easy, flexible and fun!

You don't have to be an artist or programmer, just select from Baudville's complete line of compatible graphics software. The possibilties are infinite.

Whether you've just started, or you're a seasoned hi-res hacker, you'll appreciate the unmatched power and simplicity of Baudville's graphics software.



Create motion picture shows.

TAKE 1 Stunning Animation that's fast, smooth, and flicker free!

Write, produce, and direct your own computer movies. No programming skills are needed to create superb full-color animation. With easy to use TAKE-1, you define animation shapes and movements, "shoot" scenes frame-by-frame, add text at any time, and then combine the scenes to make a complete movie. Perfect for presentations and demos.

Premiering:

- *True sprite capability for multi-layered action without additional hardware.*
- *No size restriction on animated objects.*
- *Complete editing at all stages of production. Can use hi-res pictures, shapes, and fonts from PIXIT or other graphics utilities for background scenes and animation shapes.*
- *A convenient selection of animation libraries are available.*
- *Moving objects can have "windows" that allow background to show through.*
- *Optional programmer's "tool kit" lets you put the power of TAKE-1 animation in your own programs.*

$59.95 Requires 64K

Productive tools for home, education, business, and the arts.

©Copyright 1984 Baudville

The complete color graphics package.

PIXIT

Includes these integrated features:

Create A Shape
using simple keystroke commands.

Shape Table Editor
allows up to 128 shapes placed anywhere in table.

Shape and Font Library
provides a convenient selection of pre-drawn shapes and text fonts.

Picture Editor
creates hi-res pictures using mixed text and graphics, circles, lines, colorfill, and shape tables.

Slide Show
displays hi-res pictures like a slide projector.

Printer Dump
Supports a wide variety of dot matrix printers and interface cards. No additional software required.

Uses standard hi-res picture and Applesoft shape table files which are compatible with other graphics utilities.

Optional Shape Libraries
greatly reduce the time required to generate computer art. Each library contains a wide variety of artist drawn shapes for specific areas of interest.

$49.95 Requires 48K Optional Libraries @$20 each.

For Apple II, Apple II+, Apple //e or compatible computers.

Apple is a registered trademark of Apple Computer, Inc.

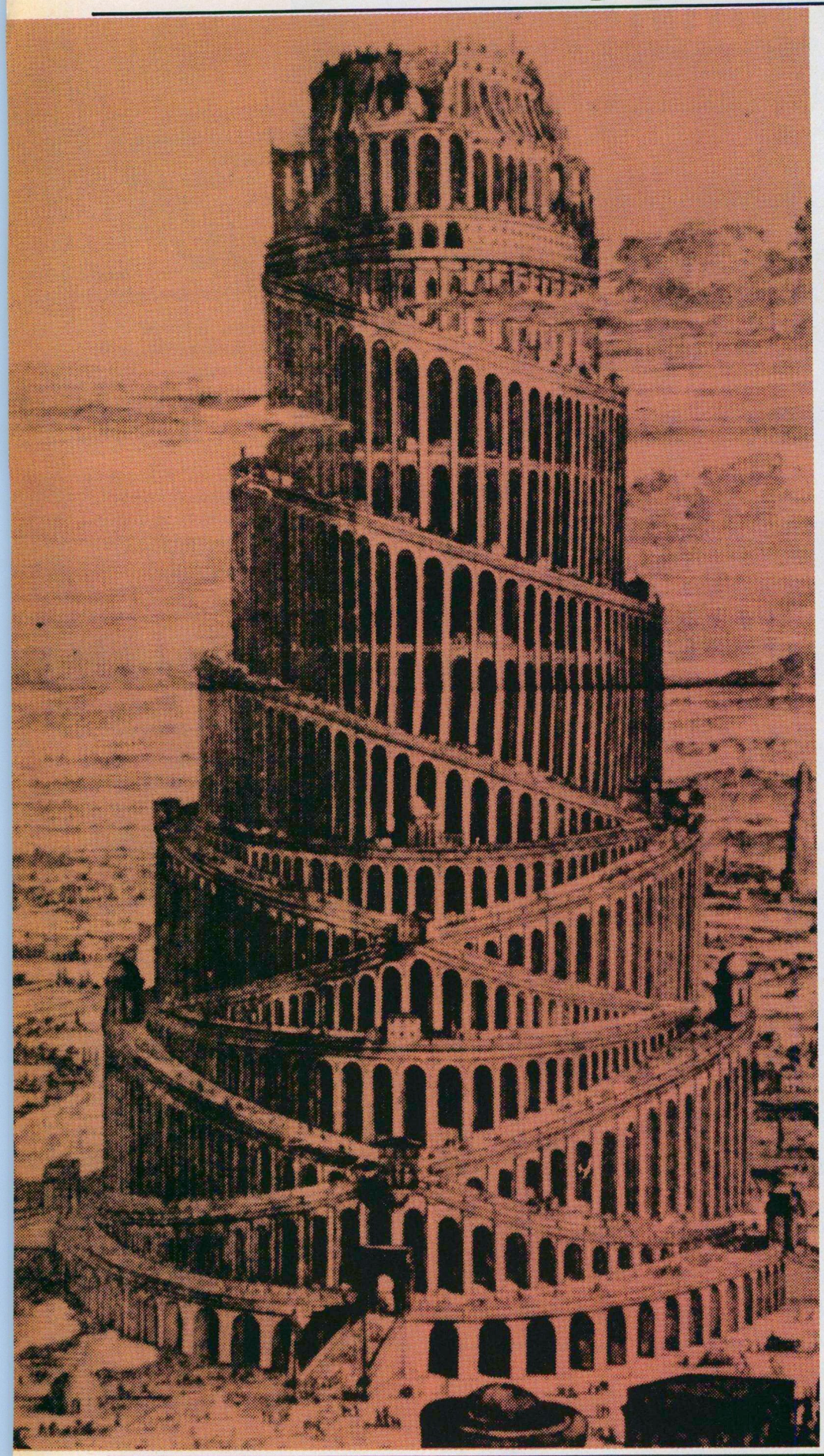
Available at your local software store or direct from BAUDVILLE.

Visa and Master Charge accepted, Michigan residents add 4% sales tax.

BAUDVILLE, 1001 Medical Park Dr., S.E.
Grand Rapids, Michigan 49506 Phone (616) 957-3036

BAUDVILLE

Circle 122 on Reader Service Card



Computer Languages

Jonathan V. Post

After the Fall of Babel

I HAVE always had problems with languages. After five years of struggling with three years of French in high school, I have nothing to show but some essays on Rimbaud (not to be confused with Kilobaud). And yet my parents and brother achieved fluency. I dropped out of Russian under the shadow of a red "F," and can only remember how to pronounce Nabokov and that "Dradnats" is "standarD" spelled backwards. It's my hidden academic weakness. But today, though tongue-tied, I can laugh—because I've written software in over *thirty* different computer languages.

My case is not uncommon. A remarkable number of computer programmers have had difficulty learning foreign spoken languages. Computer language, it seems, is a different psychological entity than natural language. Being bilingual in BASIC and English is not altogether the same as being Spanish-American. Espousing Pascal is not necessarily the mark of a Francophile.

Perhaps we should see computer languages to be closer to mathematics, or music, or chess notation. To be "computer literate" is different than to be "literate." Those who are both illiterate *and* computer illiterate may well be ravaged by unemployment. The literate and computer literate will be, however undeservingly, ruling class.

This article is excerpted from Jonathan Post's forthcoming book *Computer Futures*, to be published by Jim Baen, Inc. and distibuted by Simon & Schuster.

The "Natural" Order

Computer languages such as BASIC, FORTRAN, and Pascal are so-called "High-Order Languages" (HOL). These computer languages—like other modern languages including Esperanto, Interlingua, and Loglan—are artificial. They were created in the mid-1950s to mediate between human and machine. But there are several translation steps in this mediation. The computer instructions themselves are in binary—string after string of ones and zeros. The ones and zeros are grouped into equal-sized bit-bunches which we call bytes.

Typically the byte would be coded to mean a letter, or symbol, or digit meaningful to the human being. Two of the popular bit-byte-character codes are called ASCII (American Standard Code for Information Interchange) and EBCDIC (Extended Binary Coded Decimal Interchange Code). Primitive computer operations are then coded in terms of "instruction mnemonics" which are easier for the programmer to remember than a pattern of zeros and ones. For example, an instruction to add the number stored in one computer memory location might be coded as the mnemonic ADD, which itself can be decoded into three consecutive bytes.

Setting a memory location (or "register") to zero and adding a "packed" number to it might become ZAP—Zero and Add Packed. As John Mauchley or John Von Neumann first realized, this allows the computer program to be written as a series of numbers into computer memory, and the computer could read, follow, and modify its own "stored program" automatically.

Programming with the zero/one bytes themselves is called programming in "machine language." All of the first programmers did this, prostrating themselves at the feet of the electronic idol and praying in a series of "yea, yea, nay, nay's," yin-and-yang, positive-and negative, abstract mathematical language which was very hard to master.

Programming with instruction mnemonics is called programming in "assembly language." Being a compromise for the sake of communication, it is easier on the human—but it requires extra software. You now need a computer program called an "assembler" to translate the patterns of letters, numerals, and symbols into a list of machine instructions which the computer can digest.

Building the Tower

FORTRAN, short for FORmula TRANslation language, was developed in the 1956-57 era, the age of Sputnik. It served as a sophisticated intermediate between Assembly Language and the language of scientific and business mathematics. With FORTRAN, programming could be a more convenient mixture of made-up words and familiar equations.

COBOL (COmmon Business Oriented Language) was then marketed to use an even more English-like mode for programming business and data-processing computers. BASIC (Beginner's All-purpose Symbolic Instruction Code) was the first language developed for students and for "time-sharing"—that is, multiple terminals, each with a human user, all simultaneously programming and getting responses from a single, central computer. APL (A Programming Language) carried computer languages further into mathematical notation than FORTRAN could. LISP (LISt Processing language) simplified the manipulation of non-numerical lists and symbols. It became a standard language for artificial intelligence, where symbols are processed as knowledge units rather than numbers.

Over time, a stifling proliferation of computer languages developed. The most appropriate allegory was the confounding of tongues at the Tower of Babel. The ethnic minorities of programmers split into warring linguistic tribes, each with its own unpronounceable shibboleths. The shortage of programmers began to look like a shortage of blood types. "We're out of A,B,O and AB-." "We face an undersupply of FORTRAN 77, ADA, and C."

What does the future of computer languages hold in store? Is the hope for commonality and standardization of communication as futile today as it was in the days of Babel? For now, programmers must resign themselves to coping with each new, highly-touted language while maintaining fluency in the widely-used, existing ones. Perhaps computers themselves will someday find a solution.

Listen.
Before you buy a speech synthesizer, listen to the ECHO II. It is the most versatile, intelligible and economical speech product.
Check it out at your local dealer.
Street Electronics Corporation
1140 Mark Avenue Carpinteria, CA
93013 Telephone (805) 684-4593
Circle 68 on Reader Service Card

Bringing up FORTH

Andrew Bernat

THE programming language FORTH has been receiving considerable attention in the computer press over the last several years as a highly efficient language for control and time-critical applications. There are at least two reasons for this (beyond the eminent suitability of FORTH for such applications). The FORTH Interest Group (FIG) has placed in the public domain a standardized set of low-cost FORTH interpreter/compilers for virtually all small processors. Leo Brodie has written the first generally available beginners' level text on FORTH ("Starting FORTH," Prentice Hall; but beware—Brodie's FORTH language is not exactly the same as FIG's). FORTH is truly a language for people who like to do a job efficiently, in terms of both programming and running time.

This article does not pretend to be a FORTH tutorial—see Brodie's book. But if you wish help on bringing up your own FORTH system on your Apple, read on. Some assembly language experience is also assumed.

The FORTH language is essentially a dictionary, where each new word is defined in terms of words already in the dictionary. Only the very first few words, plus the I/O words, need to be coded in the actual assembly language for a specific machine. The code for each new word consists of a list of addresses of the previous, already defined, words used in the new word's definition. The FORTH interpreter executes a word by successively picking up the next address in the list, jumping to that address, executing the word there (by repeating the same process discussed in this sentence), returning to the original list, picking up the next address and repeating the whole process until the end of the definition is reached. You can see both types—machine code and address list—in Listing 4.

The entire FORTH system consists of a diskette with object code containing the compiler, interpreter and pre-defined words and one or more application diskettes with text definitions of new words and/or data. These new words constitute your application program.

This article describes my experiences in bringing up FIG FORTH on my Apple][, 64K, 40 column, 1 disk drive and 0 printer system. The last three deficiencies have now been rectified, but all the work described herein was done on the original system. The 64K is necessary only for ease in assembling the FORTH code. Those of you without megabuck systems are not left out!

I'll cover obtaining FIG FORTH, setting it up for Big Mac.LC, fixing a bug in the ENCLOSE word, ameliorating the Apple'S 40 column screen by using A.P.P.L.E.'s SOFT 70, modifying the disk I/O to bring it into line with Apple DOS standards and an EXEC bootup file.

Obtaining Forth and Setting It Up for Big Mac.LC

I obtained my copy of FORTH directly from FIG (P.O. Box 1105, San Carlos, CA 94070) for $15. The Apple version was written by Robert E. Kuntze and costs $15 in hardcopy listing form plus $15 more for the FIG "Installation Manual and Glossary" (which contains general information). A disk version is also available ($90 including the manual and a rudimentary editor; see also below). Incidentally membership in FIG is not required, but at $15 is very worthwhile for FORTH-devotees.

Yes, I typed in the assembly listing, line by line for 4135 lines—but I'm a pretty fast and accurate typist. (Besides, then I had more time than money.) Finding those typing mistakes which created runtime, but not assembly-time, errors certainly provided an excellent tutorial on exactly how FORTH works! In any case, make sure your copy does not hide from you the actual FORTH system as we will want to modify it. Most commercial systems won't let you touch the predefined words. All line numbers used in this article refer to the Kuntze FORTH version 1.0 dated March 1981.

The FIG version comes in three parts which were intended to be compiled separately, as TED II+ could not handle such a large project in one piece. The three parts are connected by numerous EQU(ates). So step number one is to purchase Big Mac.LC from A.P.P.L.E. and set up a main file and three auxiliary text files. The main file consists of the system constant definitions, startup literals (Part 1, lines 1 through 96) and three Big Mac.LC PUT commands to read the text files. Kuntze's Parts 1 (after line 96), 2 and 3 become the three separate text files. Use Big Mac.LC, following the manual, to accomplish this. Now delete all the EQU(ates) and etc. which Kuntze needed to link the parts together (Part 1, lines 50-65 and 1754; Part 2, lines 12-122 and 1255; and Part 3, lines 12-162).

In order to actually assemble such a large program both HIMEM and SYM must be altered in Big Mac.LC to allow space for the symbol table. Values of HIMEM:$7000 and SYM $9000 work for the version discussed in this article.

The FIG version comes with three tracing features active. This tracing is not needed, or desirable, in a working system. Each tracing mode is enabled with a single JSR. To disable them you have three options. First, you may replace each JSR in the assembly code (Part 1, lines 130, 214 and 1041) with three NOP instructions (each JSR takes three bytes; each NOP one). Second, you may remove tracing completely by deleting Part 1, lines 130, 154–262, 214 and 1041 from your assembly code. Lastly, you may generate a binary object file with the tracing active, BLOAD the file, enter the monitor, replace each three-byte location containing a JSR with three consecutive $EA's (NOPs) and then BSAVE the altered binary. The Big Mac listing tells you the locations of the three JSR's. Tracing is not compatible with SOFT 70 (see below).

Caution! It is well known that the 6502 chip does not handle indirect jumps to addresses ending in FF correctly. Such jumps are how FORTH gets around. The solution is to avoid FORTH code-field-addresses ending in FF. The FORTH Compiler does this for any new words that you define. But you are responsible for ensuring that the initial FORTH system is correctly set up. To do this NOP's are inserted to push up the code as required to avoid the FF's (e.g., Part 1, lines 88–96). As I modified Kuntze's code, I inevitably created problems. Solution number one is to search Big Mac's output and figure out how many NOP's are needed at line 96 to avoid problems. But curing one problem address tends to create problems at other addresses, so some iteration is required. Solution number two is to insert NOP's here and there, as needed, throughout the code and insert comments at line 96 pointing to the NOP's. Not as clean, but quicker.

A complete set of FORTH words to implement a 6502 assembler (in 1300 bytes!) was described by William Ragsdale in *Dr. Dobb's Journal* (September 1981, p. 12). Such an assembler is necessary for advanced FORTH work. FIG has copies available (for $7.00).

Fixing an Enclose Bug

Even after eliminating all typing mistakes (I think), my FORTH would occasionally disappear to never-never-land upon LOADing text in from disk. Before I acquired either the time or energy to search for the cause and cure, an article by Andy Biggs appeared in *FORTH Dimensions* (the FIG newsmagazine) III, p.170, detailing the problem and its solution.

The problem is in the word ENCLOSE—the FORTH text parsing primitive—which is vital to the entire FORTH system. The original FORTH Installation Manual contained William Ragsdales's 6502 (but non-Apple) implementation. Ragsdale used the Y register to index through the input text. But if the input text has a blank area greater than 256 bytes, it will overflow the 8 bit Y register resulting in an endless loop (and a trip to

The $175 disk drive for Apple II

Now, you can have a full-sized 5¼" floppy disk drive compatible with the Apple II for the unbelievably low price of $175.
Also, check out these half-height 5¼" floppies:

- Single Apple half-height $195.00
- Dual Apple half-height $390.00

(includes the drive, cabinet, cable, and our standard warranty including 90 days parts and labor)

AVAILABLE NOW FOR DELIVERY

TELEPHONE ORDERS ONLY
MASTERCARD, VISA, CASHIER'S CHECKS COD ORDERS ACCEPTED
(818) 883-0587

StarLogic

20860 Plummer Street
Chatsworth, CA 91311

Apple is a registered trademark of Apple Computer, Inc.

(Also compatible with Franklin ACE)

Circle 119 on Reader Service Card

never-neverland). This never happened to Ragsdale because his disk sectors were only 128 bytes, but on the Apple we use 256 byte sectors. Kuntze's ENCLOSE is taken directly from Ragsdale's. Biggs enclosed a revised version of ENCLOSE which implemented full 16 bit indexing. His version is repeated here as it is not generally available and is of vital importance. Replace Part 1, lines 499-534 with the code in Listing 1.

Listing 1

```
*
*  'ENCLOSE', SCR=20, L=1
*
*  TAKEN FROM BIGGS, FORTH DIMENSIONS III, P. 170
*  USES 16-BIT INDEXING
*  THE 'Y' REGISTER FORMS THE LOW INDEX BYTE
*  STACK LOCATION $1,X FORMS THE HIGH INDEX BYTE
*  THE BASE ADDRESS HELD IN N+2.  N+3 IS ALSO AFFECTED
*  N IS A GENERAL PURPOSE ZERO-PAGE STORAGE AREA
L301      HEX   87
          DCI   'ENCLOSE'
          DA    L243
ENCL      DA    ENCL+2
          LDA   #2
          JSR   SETUP
          TXA
          SEC
          SBC   #8
          TAX
          STY   $3,X
          STY   $1,X      INITIALIZE SETTING HI INDEX = 0
          DEY
          DEC   N+3
          DEC   $1,X      PRIME THESE VARIABLES FOR LOOP
L313      INY
          BNE   XXX1
          INC   N+3       INCREMENT HI ADDRESS
          INC   $1,X            AND HI INDEX
XXX1      LDA   (N+2),Y   GET CHARACTER FROM INPUT STREAM
          CMP   N         IS IT A DELIMITER?
          BEQ   L313         LOOP IF TRUE
          STY   $4,X      NON-DELIMITER SO PUT FIRST
          LDA   $1,X      RESULT ON THE STACK
          STA   $5,X
L318      LDA   (N+2),Y   GET CHARACTER AGAIN
          BNE   L327      BRANCH IF NOT A NULL
          STY   $2,X
          STY   $0,X      TIDY UP RESULTS FOR 'NULL' EXIT
          LDA   $1,X
          STA   $3,X
          TYA
          CMP   $4,X      IF FIRST AND LAST INDEXES ARE
EQUAL
          BNE   L326
          LDA   $1,X
          CMP   $5,X        THEN
          BNE   L326
          INC   $2,X              INCREMENT THIS RESULT
          BNE   L326
          INC   $3,X
L326      JMP   NEXT
L327      PHA               SAVE CHARACTER
          STY   $2,X
          LDA   $1,X        SAVE CURRENT INDEX AS OFFSET TO
          STA   $3,X        FIRST DELIMITER AFTER TEXT
          INY
          BNE   XXX5
          INC   $1,X        INCREMENT INDEX
          INC   N+3         AND HI ADDRESS
XXX5      PLA               RECOVER CHARACTER
          CMP   N           IF NOT DELIMITER
          BNE   L318           THEN LOOP
          STY   $0,X           ELSE EXIT
          JMP   NEXT
```

Ameliorating the 40 Column Headache

The FORTH language has traditionally considered any data diskette, whether consisting of new word definitions or actual data, as consisting of screens 1024 bytes long. These disk screens must be displayable for listings, editing, etc.; tradition dictates 16 lines, each 64 columns wide. And therein lies the problem for the 40 column Apple. But, once again, A.P.P.L.E. comes to the rescue (it really is amazing what a bargain membership is) with SOFT 70.

The Apple FORTH keyboard and screen text I/O are handled through the two words KEY and EMIT. These in turn work through the standard monitor hooks. Once loaded into your Apple, SOFT 70 modifies the standard hooks to allow our Apples to write 70 column-wide screens. Admittedly, not to the same quality as an 80-column card, but you can't beat the price.

First, however, some modifications are in order. I used SOFT 70 $800 as that particular region of memory is not used by FORTH anyway. The SOFT 70 manual states that zero-page locations $09, $47, $EC, $ED, $EE, $FA and $FB are used, none of which conflict with FORTH. However, things didn't work. I then disassembled SOFT 70 (using SOURCEROR, which comes free with Big Mac) and discovered that SOFT 70 $800 also uses locations $DE and $DF, which do conflict. The simplest solution was to BLOAD SOFT 70 $800, enter the monitor and change all references from $DE to $E6 (locations $8DE, $8E2, $8E5, $8EC and $8F1) and from $DF to $E7($8EA and $93C) and then BSAVE it. Further, modify the FORTH word HOME to JSR to $809 (the SOFT 70 HOME address; Part 3, line 481), COLD to JSR to $800 (the SOFT 70 initialization address; Part 2, insert after line 1022) and WARM to JSR to $803 (SOFT 70 $800 warm start; Part 2, insert after line 1044). Voila!

Now that we have a standard FORTH video screen, we need a full-screen editor. Several have been described, I chose Henry Laxen's version which was presented in *Dr. Dobb's Journal* (September 1981, p. 27; available from FIG). This editor requires that each screen be read in from disk into a 1024 byte buffer. To accomplish this, I used hi-res page 2

Listing 2

```
SCREEN #  51
  0  ( CURSOR CONTROL FOR VEDIT)
  1  CODE  70-CRTXY    ( X Y--)   ( MOVE CURSOR TO X,Y)
  2        SEC LDA,  9 STA,  BOT LDA,  FB5B JSR,  POPTWO JMP,
  3    END-CODE
  4  : 70-CLEAR-TO-EOL    ( N--)   ( CLEAR TO END OF LINE)
  5    DROP 69 2072 C@ EMIT EMIT ;
  6  : 70-CRTCLR-SCR    ( --)   ( CLEAR SCREEN & HOME)
  7    HOME ;
  8
  9
 10
 11
 12
 13
 14
 15

SCREEN #  53
  0  ( THESE DEFINITIONS MUST BE IN THE VEDIT VOCABULARY)
  1  16384 CONSTANT E-BUFFER   ( USE HIRES2 FOR THE BUFFER)
  2  : BLOCK    ( SCREEN#--E-BUFFER)
  3    E-BUFFER SWAP B/SCR * B/SCR BOUNDS DO
  4      DUP I BLOCK ( USES ORIGINAL DEFINITION)
  5      SWAP B/BUF CMOVE B/BUF + LOOP
  6    DROP E-BUFFER ;
  7  : UPDATE    ( --)
  8    E-BUFFER SCR @ B/SCR * B/SCR BOUNDS DO
  9      DUP I FORTH BLOCK ( USE ORIGINAL DEFINITION)
 10      VEDIT B/BUF CMOVE UPDATE ( USES ORIGINAL DEFINITION)
 11      B/BUF + LOOP
 12    DROP ;
 13
 14
 15
```

as a buffer—disk reads still work 1 sector at a time, but then the four sectors making up a screen are copied to the 1024 byte buffer. This is accomplished by redefining the two words UPDATE and BLOCK to not only initiate disk I/O, but to also handle the buffering. Definitions are given in Listing 2.

Three cursor control functions must also be defined—positioning the cursor to coordinates X,Y, clearing the screen to end-of-line and clearing the entire screen. These words are also given in Listing 2.

Enhancing the Disk I/O Routine

FORTH disk I/O is accomplished directly through low-level routines—there is no file structure at all. An entire diskette is considered to consist of "screens," each 1024 characters long. Thus each DOS 3.3 formatted diskette contains 140 screens numbered 0 through 139.

In Apple DOS, the low-level routine is called RWTS. "Beneath the Apple DOS" (Worth and Lechner, Quality Software) completely discusses the use of RWTS on pages 6-4 to 6-7. Kuntze's version of FORTH

New DAVID-DOS II™ Compared With Other DOS's

Now you can speedup Text-file handling two ways

DAVID-DOS II is a new edition of DAVID-DOS with added speed, commands, and features. New Read, Write and Save routines are high speed. DAVID-DOS II updates full disks like Apple's Master Create. (The programs on your disks are not touched). DAVID-DOS II Inits blank disks with Basic, Binary or Exec HELLO in seconds. Ten new commands operate identical to existing DOS commands. Use them from the keyboard or in Basic programs. They accept A & L parameters.

Ten New DOS Commands

1. **TLOAD speed loads all Text Files, random or sequential, to ram.**
2. **TSAVE speed saves all Text Files, random or sequential, from ram.**
3. **TLIST Lists all Text Files, random or sequential to screen/printer.**
4. **DUMP Memory to screen/printer in Hex with Ascii on right side.**
5. **DISA disassembles Binary to screen or printer.**
6. **AL prints last loaded program Address & Length in decimal & hex.**
7. **HIDOS moves DOS to Language Card & continues operation of program.**
8. **/ is a one keystroke Catalog in addition to the original command.**
9. **DATE prints with any clock. Also File Dating with clock or manual.**
10. **FIND prints address's of hex found in 64k memory. Hidos cmd only.**

Compatible

All DOS entry addresses have been preserved. DOS is original length and compatible with most software. David-Dos II is copyable and creates fully copyable updated disks. DAVID-DOS II is licensed by programmers for inclusion in the software they sell. Init areas were used for David-Dos II. Works with all Apple IIs including IIe 80 Col, Franklin & Hard Disks, such as Corvus & Xebec. Requires 48K. Complete documentation for screen or printing and many utilities are on the disk.

All times in seconds. (Time Test programs available)		DAVID DOS-II	ProDOS	DIVERSI DOS	DOS 3.3
TEXTFILES (100 Sectors)	TSAVE	8.0	NO	NO	NO
	TLOAD	6.2	NO	NO	NO
(791 Strings,	WRITE	29.3	28.0	29.4	88.4
32 chars ea)	READ	24.3	16.3	24.3	83.8
	PRINT/READ	44.2	45.9	45.1	117.1
(442 Sectors, 7 x 5'00)	APPEND	142.3	142.9	151.1	1231.2
APPLESOFT (100 Sectors)	*SAVE	6.4	16.4	6.4	33.1
	LOAD	5.0	4.0	5.0	23.5
INTEGER (100 Sectors)	*SAVE	6.6	NO	6.6	33.4
	LOAD	4.9	NO	4.9	23.4
BINARY (100 Sectors)	*BSAVE	7.3	18.4	7.3	28.7
	BLOAD	5.8	4.8	5.8	24.5
48K PROGRAM SPACE	APPLESOFT	36,352	NO	36,352	36,352
(With 3 Bufs avail)	INTEGER	36,352	NO	36,352	36,352
	BINARY	36,352	34,816	36,352	36,352
64K PROGRAM SPACE	APPLESOFT	46,592	31,232	45,658	35,162
(With 5 Bufs avail)	INTEGER	46,592	NO	35,162	35,162
	BINARY	46,592	40,704	45,658	35,162
NUMBER OF DOS COMMANDS		37	29	31	28
CLOCK FILE DATING		YES	YES	NO	NO
MANUAL FILE DATING		YES	NO	NO	NO
ONE KEYSTROKE CATALOG		YES	NO	YES	NO
AUTO USE INTEGER CARD ANY SLOT		YES	NO	NO	NO

Each Program was tested twice W/Apple Clock Card on a newly formatted disk containing DOS.
*Add 5 seconds for Verify. Apple II, Applesoft & ProDOS are trademarks of Apple Computer

To Order: Send Check or Phone Visa/MasterCard.
Add $2.00 Shipping. Overseas add US $4.00
Calif. add 6%. All orders airmailed.

$39.95

DAVID DATA™
12021 WILSHIRE BLVD., SUITE 212A
LOS ANGELES, CA 90025 (213) 478-7865

Circle 128 on Reader Service Card

has three deficiencies: he chose not to use the standard DOS Page 3 vectors, but rather directly accessed DOS 3.3's IOB; he did not allow for a second disk drive and he provided no error control. I chose to modify his disk I/O routines (R/W) and R/W (Part 3, lines 521–570). Listing 3 shows my modified versions of these two words and should replace lines 521–570. Note that I added the word FIOB, "find IOB," which might come in handy elsewhere.

There are two types of errors—disk range (sector not in the range 0–559) and Apple DOS error. If a DOS error occurs on drive 1, the FORTH WARNING flag is set to 0 (1 specified that error messages are available on disk) so that FORTH does not immediately attempt to access the disk to inform you. Instead you will see " MSG #8".

My development of R/W is an example of FORTH programming—always debug a new word in high level FORTH. It is not wise to immediately insert your tentative new R/W definition into the FORTH system as any bugs might destroy your data diskettes.

Listing 3

```
*  '(R/W)' MACHINE CODE ROUTINE
*
LA400       HEX  85
            DCI  '(R/W)'
            DA   LA200
PRSLW       DA   PRSLW+2
            LDY  0,X        GET IOB ADDRESS FROM STACK
            INX
            LDA  0,X                                    L3072
            INX
            STX  XSAVE                                  L3074
            JSR  $3D9       DOS VECTOR TO RWTS
            LDA  #0         B.A.D. 6-5
            STA  $48
            LDX  XSAVE
            BCS  PRSLW1     IF ERROR
            LDA  #$0        NO ERROR
            JMP  PUSH0A
PRSLW1      LDA  #$1        ERROR
            JMP  PUSH0A
*
* 'FIND IOB ADDRESS FOR APPLE DOS
*
L3080       HEX  84
            DCI  'FIOB'
            DA   LA400
FIOB        DA   FIOB+2
            STX  XSAVE
            JSR  $3E3       APPLE DOS IOB LOCATOR VECTOR
            LDX  XSAVE      PUSH ADDR ONTO STACK
            DEX
            STA  0,X
            DEX
            STY  0,X
            JMP  NEXT
*
*  'R/W' FOR APPLE DOS
*
L3060       HEX  83
            DCI  'R/W'
            DA   L3080
RSLW        DA   DOCOL
            DA   FIOB       PUSH IOB ADDR
            DA   TOR             TO RETURN STACK
            DA   ZERO       VOLUME NUMBER
            DA   LIT
            HEX  0300       --OFFSET
            DA   R
            DA   PLUS
            DA   CSTOR
            DA   TWO        OPERATION
            DA   SWAP
            DA   SUB
            DA   LIT                                    L3076
            HEX  0C00       -- OFFSET
            DA   R
            DA   PLUS                                   L3078
            DA   CSTOR
            DA   DUP        BLOCK RANGE CHECK
            DA   ZLESS      --CHECK >= 0                L3082
            DA   ZEQU
            DA   SWAP       --CHECK <= MAX
            DA   DUP
            DA   OFSET
            DA   AT
            DA   LIT
            DA   SECTR      --SECTORS/DISKETTE
            DA   PLUS
            DA   LESS
            DA   ROT
            DA   ANDD
            DA   ZBRAN      --IF BAD                    L3084
            DA   L3082-*
            DA   OFSET      DRIVE NUMBER
            DA   AT
            DA   DDUP
            DA   ZEQU
            DA   ZBRAN      --IF DRIVE 2
            DA   L3072-*
            DA   ONE
            DA   BRAN
            DA   L3074-*
            DA   SLMOD
            DA   ONEP
            DA   TWO        --OFFSET
            DA   R
            DA   PLUS
            DA   CSTOR
            DA   LIT        TRACK & SECTOR
            HEX  1000       --SECTORS/TRACK
            DA   SLMOD
            DA   LIT
            HEX  0400       --TRACK OFFSET
            DA   R
            DA   PLUS
            DA   CSTOR
            DA   LIT
            HEX  0500       --SECTOR OFFSET
            DA   R
            DA   PLUS
            DA   CSTOR
            DA   DUP        BUFFER ADDR
            DA   LIT
            HEX  0800       --OFFSET
            DA   R
            DA   PLUS
            DA   STORE
            DA   RFROM
            DA   PRSLW      GO READ THE DISK
            DA   ZBRAN      IF NO DOS ERROR
            DA   L3078-*
            DA   TWO        --MARK BUFFER AS UNUSED
            DA   SUB
            DA   ZERO
            DA   SWAP
            DA   STORE
            DA   FIOB       SET WARNING FLAG TO NO DISK
            DA   TWO        --OFFSET
            DA   PLUS
            DA   CAT        --GET DRIVE NUMBER
            DA   ONE
            DA   EQUAL
            DA   ZBRAN      --BUT NOT IF IT WAS DRIVE 2
            DA   L3076-*
            DA   ZERO
            DA   WARN
            DA   STORE
            DA   LIT
            HEX  0800       --ERROR NUMBER
            DA   ERROR      --NO RETURN FROM ERROR
            DA   DROP       TO HERE IF NO DOS ERROR
            DA   BRAN
            DA   L3084-*
            DA   DROP       TO HERE IF BLOCK RANGE ERROR
            DA   RFROM
            DA   DROP
            DA   TWO        --MARK BUFFER AS UNUSED
            DA   SUB
            DA   ZERO
            DA   SWAP
            DA   STORE
            DA   LIT
            HEX  0600       --ERROR NUMBER
            DA   ERROR
            DA   SEMIS
*
```

Listing 4

```
SCREEN #  75
 0  ( R/W TESTING)
 1  CODE FINDIOB  ( PUSH IOB ADDRESS TO STACK )
 2       XSAVE STX,  3E3 JSR,  XSAVE LDX,  DEX,  0 ,X STA,
 3       DEX,  0 ,X STY,  NEXT JMP,  END-CODE
 4  0  VARIABLE DOSERROR   ( SIMULATES DOS ERROR)
 5  : DOSERROR! DOSERROR ! ;
 6  : .U ( UNSIGNED PRINT)  S->D DROP 0 D. ;
 7  ( REDEFINITIONS FOR TESTING PURPOSES)
 8  : !  .U .U ;  : C!  .U .U ;  : (R/W) .U DOSERROR @ ;  -->
 9
10
11
12
13
14
15

SCREEN #  76
 0  ( R/W TESTING CONTINUED)
 1  HEX
 2  : R/W  ( TEST R/W BY PRINTING OUT RESULTS)
 3            ( ON STACK ARE: BUFFER ADDR, BLK #, OPERATION)
 4    FINDIOB  >R  ( GET IOB ADDR & PUSH TO RETURN STACK)
 5    0 03 R + C!  ( VOLUME NUMBER)
 6    2 SWAP -  0C  R + C! ( OPERATION TO PERFORM)
 7    DUP 0< 0= ( CHECK THAT BLOCK NUMBER > 0)
 8    SWAP DUP OFFSET @ 230 ( SECTORS/DISKETTE) + < ( AND < MAX)
 9    ROT AND   ( NOW 1 ON TOP OF STACK IF BLOCK OKAY)
10    IF  OFFSET @ -DUP   ( CHOOSE DRIVE NUMBER)
11      IF 1 ELSE /MOD 1+ ENDIF  2 R + C!
12        10 ( SECTORS/TRACK) /MOD  04 R + C!  ( TRACK NUMBER)
13                                  05 R + C!  ( SECTOR NUMBER)
14        DUP  08 R + !  ( BUFFER ADDR)
15  -->

SCREEN #  77
 0  ( R/W TESTING CONTINUED)
 1        R>  (R/W)   ( GO READ THE DISK)
 2        IF  ( DOS ERROR)
 3          2 - 0 SWAP !   ( MARK BUFFER AS UNUSED)
 4          FIOB 2 + C@ 1 =   ( WARNING FLAG = 0 IF DRIVE WAS 1)
 5          IF 0 WARNING ! ENDIF
 6             08 ERROR  ( ERROR NUMBER=8; NO RETURN FROM ERROR)
 7        ELSE  ( NO ERRORS) DROP ENDIF
 8    ELSE  ( RANGE ERROR)
 9       DROP R> DROP 2 - 0 SWAP !  06 ERROR  ENDIF
10       ;

11  DECIMAL
12
13
14
15
```

First, I redefined the storage words ! and C! to simply print out the storage address and data value (rather than actually storing). This redefinition does not affect previously defined words—they continue to use the original ! and C! definitions. I also redefined the low-level routine (R/W) to simulate either an error-free or a DOS error return (through the use of the variable DOSERROR). I then defined my new R/W routine in FORTH (see Listing 4). By executing this new R/W I was able to see what it would do—if I were to let it. When R/W was working properly (which did not occur immediately), I converted it into assembly form and used Big Mac to edit it into the FORTH system (as shown in Listing 3).

Additionally, I added four more disk buffers by changing the following system constants: NBUF to 8 (line 26), BMAG to $820 (line 29), MEM to $9A10 (line 39) and UAREA to $9990 (line 40).

EXEC File Booting

To run FORTH you need to collect two files on the same diskette: the FORTH binary object code from Big Mac and the modified SOFT 70 $800 binary object code. BLOAD SOFT 70 $800 and BRUN FORTH—you're up and running. A fancier approach is to have your HELLO file do the work for you. You can't simply use a simple Applesoft program containing the DOS commands, because the Applesoft program conflicts with SOFT 70 $800. Instead, run the Applesoft program given in Listing 5 which creates a text EXEC file containing the proper commands. Put this text file on the same disk with your binary object files. Then EXEC this file to get going. To be really fancy, modify DOS to use your EXEC file upon booting up (Beneath the Apple DOS, page 7-3).

Remember to remove your system disk after booting and replace it with your text screens disk. Also, do not forget to type EMPTY-BUFFERS to clean out the disk I/O buffers before your first disk command.

Listing 5

```
10  D$ = CHR$(4) : REM  DOS CNTRL-D
20  REM  CREATE EXEC FILE
22       PRINT D$"OPEN EXEC.FORTH"
24       PRINT D$"WRITE EXEC.FORTH"
30  PRINT "BRUN SOFT SEVENTY $800"
40  REM  SET UP RESET VECTOR BACK TO FORTH
42       PRINT "POKE 1010,04" : REM 9704 = $6104
44       PRINT "POKE 1011,97"
46       PRINT "CALL -1169"
50  REM  PRINT LOAD MESSAGE
52       PRINT "PRINT CHR$(76);CHR$(79);
                CHR$(65);CHR$(68);CHR$(73);
                CHR$(78);CHR$(71);CHR$(32);
                CHR$(70);CHR$(79);CHR$(82);
                CHR$(84);CHR$(72)"
60  PRINT "BRUN FORTH"
70  PRINT D$"CLOSE EXEC.FORTH"
80  END
```

Speaking of Tongues:

A Review of Foreign Language Learning Software

Kaiulani Schuler

IF YOU'VE ever experienced the frustration of learning or teaching a foreign language in a classroom, perhaps the idea of computer-assisted instruction (CAI) appeals to you. Learning a foreign language involves so many different levels of cognitive processing that a classroom situation can easily fall short of providing the needed emphasis of each concept relative to individual needs. Current computer technology cannot match a human teacher in providing qualitative feedback to oral exercises in pronunciation, comprehension, or spontaneous verbal production. However, the feedback capabilities of a computer offer the exciting potential to customize and channel learning based on individual strengths and weaknesses.

There are many foreign language learning packages available for the Apple. Some are designed to be supplemental aids for the instructor while others are directed at the Apple user who wishes to learn at home. This review will look at examples of both. For each software product, I will first describe its objectives, functions, and capabilities. Then I'll present an evaluation based on three criteria: content, presentation, and feedback.

Clearly, a lesson's contents should be relevant and useful. It is important that these contents are reviewed and tested in a manner that maximizes learning and comprehension of the material. Mastering a foreign language requires hours of practice and repetition. Thus, material should be presented in a variety of forms to avoid boredom and to reinforce a concept in memory. Examination drills should complement these forms to further reinforce learned material. These are the criteria of a good workbook. Educational software should be more than a glorified workbook. The interactive nature of CAI should be used to the student's best advantage by providing helpful feedback.

There are several levels at which CAI can provide feedback to the student. The computer can . . .

1. announce if an answer is correct or incorrect, correct errors and provide a total score of performance on a given section.
2. give an analysis of each error.
3. provide a diagnostic analysis of the types of errors that occur over an entire section and advise an appropriate course of study.

Most educational software provides at least the first level of feedback, but only the more sophisticated programs approach the second or third levels as a good human tutor would.

Skill Drills

La Guillotine
Target language: French
Level: Beginning—Intermediate
For: Apple][+, //e
Price: $29.95. Copy-protected.
Company: Gessler Educational Software

This is a self-paced vocabulary-building program with five categories and two levels of words (easy and hard). The menu offers three options: review and study vocabulary, matching exercise, or *La Guillotine* (a variation of the "hangman" game). After selecting a word set, the student is presented with 10 French and English equivalents to study.

In the matching exercise, the English and the French words are presented in lists and the student must match the equivalents. Errors are marked on the screen at the completion of the quiz. *La Guillotine* utilizes graphics and sound to set the scene in revolutionary France. An unfortunate aristocrat in tails lays with his head across the block of a guillotine while a basket waits next to a jeering crowd. The student must guess at the letters contained in a word from the word list. Five incorrect guesses are allowed before the guillotine falls and the frowning head tumbles. If the correct spelling is achieved, however, the aristocrat grins with relief and a few bars of "La Marseillaise" are played. Words missed are presented at the completion of both exercises and the student is urged to study them.

As a supplemental teaching aid, this program excels in one area—it employs a clever device to make learning fun, something that is always desirable for unenthusiastic or very young students. However, I doubt that it would hold the attention of an older, intermediate student for long. The matching exercises are not very useful, as they involve simple recognition of the items studied and it is possible to match them correctly without really learning them. Knowledge of spelling is required for "La Guillotine," but this knowledge is not reinforced in a matching exercise as it would be in sentence completion exercises.

La Carte de France
Target language: French
Level: Intermediate—Advanced
For: Apple][+, //e
Price: $47.95. Copy-protected.
Company: Gessler Educational Software

The objective of this program is to introduce students to the geography, history, industry and culture of the regions and areas of France. There are

14 sections corresponding to the 14 regions of France. Each section contains a description of one of the regions, followed by a quiz containing 5 questions. The student is free to select any section and to review each page of text prior to taking the quiz. There are four types of quizzes: true–false (vrai–faux), matching, sentence completion in which the correct answer is provided in a scrambled form, and sentence completion in which the correct answer must be selected from a list of three choices. There is a final exam of the latter form with 14 questions (one from each section). The entire program is in French, except for explanations of the quiz format.

This program provides excellent practice in reading comprehension through the presentation of interesting facts about France. In the sentence completion exercises, it provides feedback on the first level, i.e., it announces errors and, after the second wrong attempt to answer a question, insists the correct answer be typed in correctly before continuing. Thus, the student must think about the actual item being learned. However, I would like to see a differentiation between spelling errors and incorrect choices in the feedback to the student. A glossary that the student could access from the review section would also be useful.

Le Present et le Passe Compose
Target language: French
Level: Beginning—Intermediate
For: Apple][+, //e
Price: $35.95. Copy-protected.
Company: Gessler Educational Software

The objective of this program is to teach and review the formation of the present and past participle tenses of regular and irregular French verbs. There are six sections: the *Present* tense of regular verbs (*–er, –ir,* and *–re*), the *Present* tense of irregular verbs, the *Passe Compose* with *avoir,* and the *Passe Compose* with *etre*. Each section contains a review of the respective verb conjugations followed by a sentence completion exercise of ten questions. If the first response is incorrect, one of nine symbols appears, indicating the type of error. If the second answer is incorrect as well, the answer is given.

This program provides level 2 feedback with the coded analysis of errors. Isn't it nice to have a friend say: 1) "you're wrong, here's the type of error you've made...now fix it" and 2) "you're wrong again, here's the answer, type it in correctly and Don't Forget It!." If only it said, at the completion of a section, "In looking at your errors, it seems you have a problem grasping concept X. Why don't you consult disk Y which covers that concept in detail." If only there were more exercises per section. The review sections utilize clever graphics to focus attention on the formation of the verbs.

Soflight Language Series
Target language: Italian
Level: Beginner—Intermediate
For: Apple][+, //e
Price: $56.95.
Not copy-protected.
Company: Soflight Software

This is a vocabulary learning program with 1096 Italian–English pairs stored on disk. You must create your own study file (of between 20 and 40 words) from this list for storage on a separate disk. The menu offers the choices of reviewing the word list in either alphabetic or random order, or taking a quiz that is randomized. In the quiz, the Italian word is presented and you must type in the English equivalent. Answers are scored immediately and a cumulative score is presented when desired.

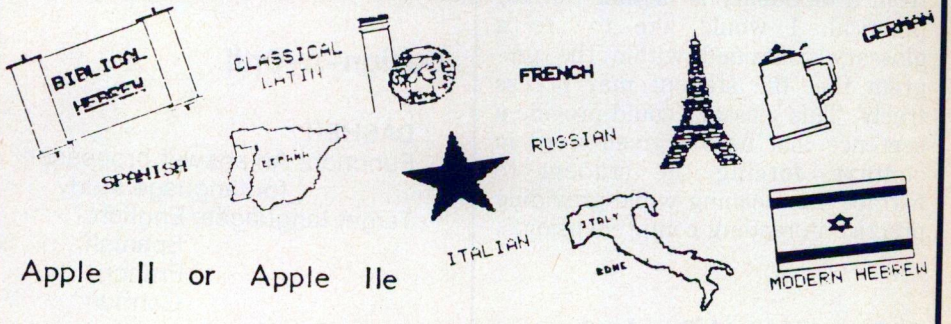
Rote memorization of word pairs is not, in my opinion, the ideal way to learn vocabulary. The use of a word in context is necessary to establish a concept in memory. I would like to see other types of exercises using the same vocabulary interspersed with this exercise—something to break up the monotony and to reinforce understanding through different contexts. The freedom to create files is a plus as it allows the student the opportunity to choose study material. There are some problems within the program as far as answer acceptance. It will accept an answer, regardless of how complete it is, as long as the letters typed in are in correct sequence. For example, the verb *abitare* is defined as "to dwell;live." The program will accept as answers: "t," "to," "to d," "live", "l," "to dwell," etc. but not "to live".

SOFLIGHT SOFTWARE

CONTACT:
Soflight Software
2223 Encinal Station
Sunnyvale, CA 94087
(408) 735-0871

P.O. BOX 2223 ENCINAL STATION SUNNYVALE, CALIFORNIA 94087 (408)735-0871

FOREIGN LANGUAGE DEVELOPMENT SYSTEM



Apple II or Apple IIe

Languages in the product line include
- Biblical Hebrew and Modern Hebrew
- Polish
- Swedish
- Classical Greek
- Arabic
- Spanish
- French
- German
- Italian
- Latin
- Russian

retails for $56.95.

Circle 58 on Reader Service Card

Augmentez Votre Vocabulaire
Target language: French
Level: Intermediate—Advanced
For: Apple][+, //e
Price: $34.95. Copy-protected.
Company: The Professor

Designed to build vocabulary, this program consists of 10 lessons of 40 problems each. The first five lessons concentrate on synonyms and the remaining five deal with antonyms. There are no study/review sections—only multiple-choice exercises. The object of these exercises is to choose a word from a list of five that is either most similar or opposite in meaning (depending on the lesson) to a given word. A performance score is presented upon exiting the program. This score is based on the number of successful answers out of the total number of attempts (as opposed to the total number of problems).

The words presented in these exercises are fairly difficult although many are English cognates and can be guessed at. They are words that are well worth adding to a student's vocabulary. However, the completion of these exercises will not, in my opinion, enable a student to learn them. Only a student who has been previously exposed to many of them could expand his word knowledge through this type of presentation. The instructions state, "When you don't know the meaning of a word, look it up in a dictionary." This approach suggests to me that the full potential of CAI is not being utilized. In my opinion, learning a language from a dictionary is neither fun nor practical. I would like to see a glossary contained within the program that the student may access freely. This glossary could provide a sentence that uses a given word in context, forcing the student to extract the meaning while providing practice in reading comprehension.

A Coordinated Curriculum

Lecciones de Espanol
Target language: Spanish
Level: Intermediate
For: Apple][+, //e
Price: $110.00. Copy-protected.
Company: CONDUIT

Lecciones de Espanol is a package containing a Teacher's Guide in a three-ring binder, a Student Reference Card, two program diskettes and two back-up diskettes. The two disks contain 7 lessons: para vs. por, ser vs. estar, preterit vs. imperfect, and 4 lessons on the use of the subjunctive. Each lesson starts with an introduction that offers instructions or reviews if desired. The lessons are actually drills in several different formats: sentence completion, transformation exercises, sentence translation, true-false, and multiple-choice. There are hundreds of questions contained in these drills and they are usually presented in blocks of 15–20 with the option to continue. Informative feedback is provided on both correct and incorrect answers. When possible, the nature of an error is analysed and explained so that the student can figure out the correct response. The program contains 5 commands that the student may use to direct his course of study or request different types of information. He may: skip to another section; stop at any point and retrieve his score; or request the answer, the definition of a word, or help in understanding a concept.

Lecciones de Espanol excels in both content and presentation. Material is consistently presented in a clear, concise, and complete manner. The diversity of the drills, the freedom to move around within the program, and the interactive analysis of errors ensure that the student comprehends the material without becoming bored.

Design-A-Drill

DASHER
Function: An answer processor for language study
Target languages: English, Spanish, French, German
For: Apple][+, //e
Price: $150.00.
Not copy-protected.
Company: CONDUIT

DASHER contains a User's Guide and four diskettes containing programs, text materials, and character sets for creating computer-assisted drills in English, Spanish, French, and German. Instructors supply questions, answers, instructions, messages, and examples. The program, DANCER, allows the instructor to specify drill format and answer-acceptance parameters. The standard drill format is full-sentence question and answer but can be modified for other formats and variations such as transformation, substitution, construction, fill-in, true-false, multiple-choice, sentence unscrambling, translation, and vocabulary drills. It is also possible to execute dictation exercises with the aid of a cassette player connected to the computer. Errors are replaced by dashes with the cursor positioned over the first error. The student must then type in the corrections.

DASHER, and programs like it, throws the door to computer-assisted instruction wide open. Following the widespread penetration of microcomputers into every-day life, DASHER makes it possible for a user at any level of computer expertise to custom design educational software. That's exciting!

The Great Creator
Function: a multi-lingual questionnaire generator
For: Apple][+, //e or Apple compatible computer
Price: $399.95.
Not copy-protected.
Company: The Professor

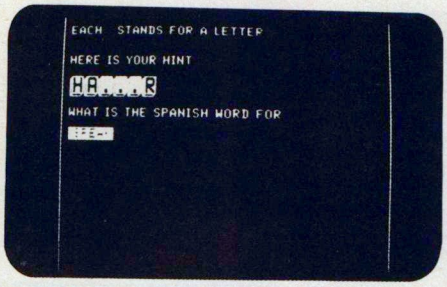
The Great Creator can be used in 17 different languages to create multiple-choice, true-false, and fill-in-the-blank drills in any subject. Character sets for additional languages and special math symbols are also available. No additional hardware is required to produce special characters as the multi-lingual software is supposed to be compatible with most printers. The text entry program contains a word processor which allows up to 450 characters for a question and 225 for an answer choice. Each choice can result in the display of a specific response, offering feedback of the instructor's choosing. It is possible to get hard copies of student scores, questions, choices, responses, or simply the questions and choices in a written quiz format. The latter option is useful for instructors who might want to construct an exam on The

Great Creator but wish to give it to several students simultaneously.

The Great Creator differs from DASHER in that it can create drills in virtually any discipline in virtually any language. However, DASHER offers more choices in quiz design.

Interactive Audio

Lamplighter Language Lab
Target language: Spanish
Level: Beginning—Intermediate
For: Apple][+, //e or IBM PC
Price: $235.00.
Not copy-protected.
Company: Lamplighter Software, Inc.



This system consists of an interactive audio cassette interface and a large, three-ring binder that contains documentation, six cassettes, and two disks. The cassette interface is connected to the computer via the Game I/O socket and plugs into a cassette player's REMOTE socket. The primary purpose of the cassettes' interaction with the software is to provide the instruction in pronunciation that a computer can't. There are five lesson sections offered on the main menu: Vocabulary, Verb I, Verb II, Special Lessons, and Conversation. Each section consists of 10 lessons containing 15 to 20 items and is presented in Study, Quiz, and Pronunciation subsections. The Study section requires the choice of presentation speed—slow, medium, or fast. English words scroll across the screen, transforming into Spanish midway. The student is immediately quizzed before studying the next item. The Quiz section requires typed answers and offers hints, if desired, upon making an error. It fills in the letters one at a time until a correct response is given. At the completion of the quiz, the student's performance is logged on disk. The Pronunciation section couples the presentation of an item on the screen with audio pronunciation on cassette. REPORTS is a program that reviews a student's progress over the course and displays it in both listing and graph form. A print-out of either is possible. CUSTOM CURRICULUM is a program available separately that is intended for use with the Language Lab. This program enables instructors to include lessons of their choosing in the Lab presentation. I'm not sure, however, that it would allow the use of cassettes.

Clearly, the unique feature of this system is that it offers audio instruction in conjunction with computer-assisted exercises. The pairing of auditory and visual material is crucial to learning a foreign language. The inclusion of this feature in computer/student interaction pushes CAI one step closer to approximating ideal human teacher/student interaction. The next step is to develop the technology that would make it possible for a microcomputer to analyze a student's pronunciation and provide diagnostic feedback. There are still a few bugs that must be worked out in the current system. When using the cassette with the software, it is not possible to exit or skip sections freely. The presentation order cannot be randomized each time. The manual warns that a high-quality cassette player must be used to ensure synchronicity. Unfortunately, my machine must be less than high-quality as the presentation was unsynchronized at times.

The error feedback this program offers is interesting. It involves the student in an interactive guessing game that focuses attention on spelling and construction.

List of Companies

Gessler Educational Software
900 Broadway
New York, NY 10003

Soflight Software
2396 Encinal Station
Sunnyvale, CA 94087

The Professor
P.O. Box 301
Swanton, Vermont 05488

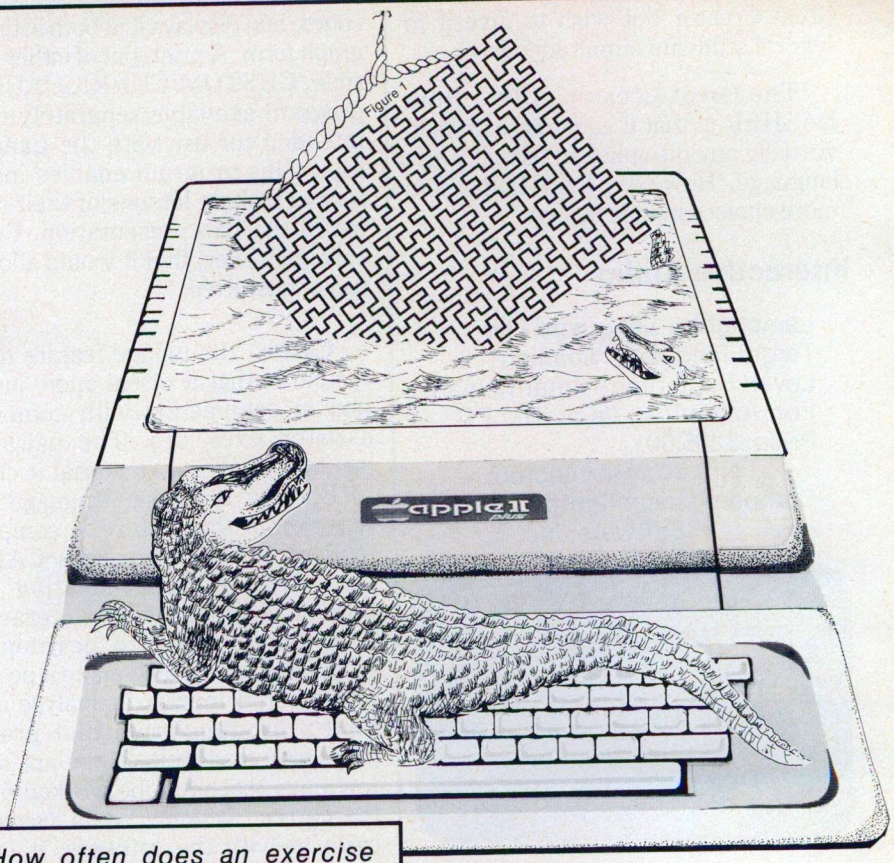
Conduit
University of Iowa
Oakdale Campus
Iowa City, Iowa 52242

Lamplighter Software, Inc.
7 Breton Ave.
Melville, New York 11747

? QUESTIONS? ?
? The Great Creator Multi-Lingual ?
? Questionnaire Generator ?
? "...like the Cadillac of the fifties: ?
? smooth and elegant and mostly automatic, ?
? the ultimate in user-friendly..." ?
? Softalk: February 1984 ?
? The Great Creator is a multi-lingual ?
? multiple-choice, true or false and fill- ?
? in-the-blank questionnaire generator. ?
? Works in 16 different languages in ?
? addition to English without additional ?
? hardware! Includes print option. ?
? Choose the French expression that ?
? correctly translates the English ?
? phrase (in parentheses) in order to ?
? complete the following sentence: ?
? Lui et moi ...(met) à Paris l'été ?
? passé. ?
? 1)se sont rencontrés ?
? 2)ayons rencontré ?
? 3)s'ont rencontré ?
? 4)nous nous sommes rencontrés ?
? 5)nous nous avons rencontrés ?
? This is not the correct French ?
? expression. The auxiliary verb ?
? 'avoir'(avons) should not be used ?
? here. In this case, être is the ?
? appropriate auxiliary verb. ?
? Your choice? 2 ?
? Questionnaire Generator ?
? With QUESTIONNAIRE GENERATOR, educators ?
? can easily create and operate any type ?
? of English language questionnaire on ?
? the computer. This program gives you ?
? the ability to follow your own curriculum ?
? and needs. ?
? Foreign Language Questionnaire Generator ?
? Offers the same features as the ?
? Questionnaire Generator program plus the ?
? ability to function in a foreign language. ?
? Math & Science Questionnaire Generator ?
? A breakthrough for math and science ?
? educators! Create your own scientific ?
? questionnaires, exams, drills and other ?
? classroom work using scientific formulas ?
? on the computer. The special symbols for ?
? physics, chemistry, calculus, etc. can now ?
? be included in your text and displayed on ?
? the screen! ?
? A solution of $Fe_2(SO_4)_3$ has a molar ?
? concentration of 0,5 mol/l. What is ?
? the molar concentration of the ?
? SO_4^{2-} (aq) ions? ?
? $Fe_2(SO_4)_3$ (S) → $2Fe^{3+}$ (aq) + $3SO_4^{2-}$(aq)? ?
? 1) 0,5 mol/litre ?
? 2) 1 mol/litre ?
? 3) 1,5 mol/litre ?
? 4) 2 mol/litre ?
? 5) None of the above. ?
? Excellent! The molar concentration ?
? of the SO_4^{2-} ions is 3 times that of ?
? the $Fe_2(SO_4)_3$ solution according to ?
? the balanced equation which is given. ?
? Your choice: 3 ?
? THE PROFESSOR ?
? 959 N.W. 53rd Street, Ft. Lauderdale, FL ?
? 33309-9990 [800] 222-1399 [305] 771-6498 ?
? Telex no. 522-600 AAA Exect Mar A59 ?
? FREE BROCHURE ?
Circle 56 on Reader Service Card

Hilbert and the Alligators

or How a Persistent Programmer Developed a Recursive Graphics Benchmark



How often does an exercise that was started for fun lead to a useful outcome? In this saga, a member of A.P.P.L.E.'s Board of Directors relates how a quest for interesting computer graphics became a race against time.

Don Williams

Once Upon A Time . . .

I CAN RECALL, with some amazement, the joy of watching Apple Pascal screaming through a display of the Hilbert space filling curve (Figure 1). This shape can be generated in varying degrees of convoluted complexity (or "orders") by a fairly simple, recursive algorithm. The Integer BASIC graphics routines seemed much slower than APPLESOFT or Pascal. I attributed much of the observed speed to the lightningly fast UCSD Pascal pseudo code interpreter. What could be faster than the best that Apple had to offer?

Some years later I acquired a copy of Ap.Graphics the Filler. The Filler is an APPLESOFT extension program written by James McBride to fill "bounded areas" on the high resolution (hi-res) graphics screen. Again, I was fascinated by this wonder that ran before me. One of the demo programs sent with the Filler was called the MAZE. The MAZE is marvelous. First it builds a 40 by 40 maze and then it uses a single call to the Filler to fill the maze. This gets quite complicated, as the Filler appears to be filling all the paths at once.

The Plot Thickens

After seeing the MAZE run a few times, I wondered how well the Filler would do with a Hilbert curve. However, the program that draws the Hilbert curve is in Pascal and the Filler is intended to interface with APPLESOFT. The first solution that came to mind was to disassemble the Filler, convert the disassembly to a text file in DOS 3.3 format, convert the DOS 3.3 format to Pascal disk format, read it in to Pascal, convert the APPLESOFT interface to a UCSD Pascal interface, modify the HILBERT program, compile the new HILBERT program, assemble the Pascal version of the Filler program, bind them together, and finally debug the whole mess.

Upon careful consideration I decided to find a better way. Converting the HILBERT program to APPLESOFT seemed a whole lot easier. After printing out a copy of HILBERT from Apple's Pascal 3 Demo disk (Listing 1), I went about the translation.

As you might expect, there were a couple of little problems. The Pascal program counted rather heavily on Turtle Graphics, one of the features of UCSD Pascal, and a little known (but much misused) feature know as recursion. In the recursion process, a program changes its arguments and calls itself to answer the question it was just asked. Unfortunately, the structure of APPLESOFT is not well suited to true recursion. That little problem was solved by saving the variables that are used to set the arguments before the program can change them, and putting them back before leaving the recursive program. An example of this can be seen in lines 210–212 and 260–262 of the BASIC versions.

Wrestling With the Turtle

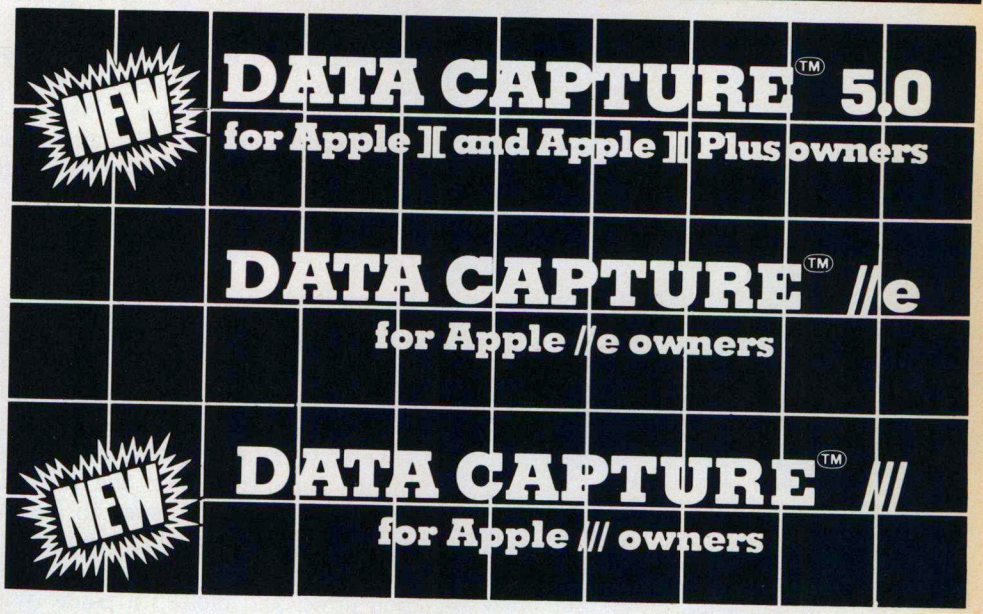
To deal with the turtle-oriented nature of the Pascal version, I used a set of routines known as TURTLESOFT (Listing 2). TURTLESOFT is an APPLESOFT extension program allowing the use of "Turtle Graphics" from APPLESOFT ampersand (&) calls. For all but the most trivial examples of a Hilbert curve the Pascal was faster, hands down. This is as it should

Listing 1

```
PROGRAM HILBERT;

USES TURTLEGRAPHICS;

VAR SIZE,DELTA,N: INTEGER;
    ORDER: INTEGER;
    CH: CHAR;

PROCEDURE HIL(I:INTEGER);
VAR A,B: INTEGER;

  PROCEDURE HIL1;
  BEGIN
    TURN(A); HIL(-B); TURN(A);
  END; (* HIL1 *)

  PROCEDURE HIL2;
  BEGIN
    MOVE(SIZE);
    HIL(B);
    TURN(-A); MOVE(SIZE); TURN(-A);
    HIL(B);
    MOVE(SIZE);
  END; (* HIL2 *)

BEGIN (* HIL *)
  IF I=0 THEN TURN(180)
  ELSE
  BEGIN
    IF I>0 THEN
      BEGIN
        A:=90; B:=I-1;
      END
    ELSE BEGIN
      A:=-90; B:=I+1;
    END;
    HIL1; HIL2; HIL1;
  END;
END; (* HIL *)

BEGIN (* MAIN PROGRAM *)
  WRITELN('WELCOME TO HILBERT');
  WRITELN('ENTER ORDER 0 TO QUIT.');
  REPEAT (* FOR EACH ORDER *)
    WRITE('ORDER: '); READLN(ORDER);
    IF (ORDER < 1) OR (ORDER > 7) THEN
      BEGIN
        WRITELN('THAT''S ALL FOLKS...');
        EXIT(PROGRAM);
      END;
    INITTURTLE;
    DELTA:=2;
    (* CALC STARTING X,Y AND SIZE *)
    FOR N:=2 TO ORDER DO DELTA:=DELTA*2;
    DELTA:=DELTA-1;
    SIZE:=190 DIV DELTA;
    DELTA:=(DELTA*SIZE) DIV 2;
    MOVETO(140-DELTA,96-DELTA);
    PENCOLOR(WHITE);
    HIL(ORDER);
    (* WAIT FOR KEYSTROKE,THEN SWITCH
       TO TEXT *)
    READ(KEYBOARD,CH); TEXTMODE;
  UNTIL FALSE;
END.
```

be when comparing an interpreted language to a pseudo-coded language. I had made a programming decision to use a set of routines that would make my programming time shorter, but might impact the amount of memory or the speed adversely. Since the program I was writing was relatively small, I didn't need to worry much about the size. Speed, however, is important, as it separates the "can-do's" from the "can't-cut-its" in programming.

What do over 20,000 Apple owners have in common? They are using **DATA CAPTURE**™ as their modem program. And time after time they recommended it to their friends. Or the company they work for suggests they buy it. And they voted it #31 of all Apple software in the 1982 Softalk reader poll. And it has been on the Softalk best seller list since 1981. If you are buying a modem program and you don't take a look at **DATA CAPTURE**™ you could be cheating yourself. Maybe there is user friendly modem software. And a 'friend of the user' company to stand behind it.



- **300/1200 Baud Operation** - for use with all popular information and timesharing systems plus other microcomputers, minicomputers, and mainframes.
- **40/80 Column** display on //e. 40 column or Videx display on Apple][with **DATA CAPTURE**™ **5.0**.
- **Edit Feature** - for preparing text to send to another computer or for editing text you have received from another computer. Many **DATA CAPTURE**™ owners use the editing feature off-line for preparing letters and/or performing other short typing tasks.
- **Capture Buffer** - You can List, Save, Print, or Transmit all or part of the Capture Buffer. The most flexible use of captured data of any program on the market.
- **One Key Commands** - output logon messages, passwords, and system commands with a single keystroke.
- **Menu Selections** - You don't have to learn complicated commands used by other types of modem programs. (Once you become accustomed to the menus you can skip them by using the Fast Menu feature.)
- **NOT Copy Protected** - We trust you.
- **Buyer Support** - We are a small company and we depend on your good will for our business. We have a five year history of full support for purchasers of our programs.
- Not sure what modem hardware to buy? - Call us for advice. For over 5 years we have been helping Apple owners put together the right communications software and hardware.

Ask your dealer for **DATA CAPTURE**™ or buy direct. We usually ship the same day and you do not have to wait for your personal check to clear.
We accept Visa and MasterCard.

Price - **$90.00.**

Software designed with you in mind.

Southeastern Software

7743 Briarwood Drive
New Orleans, LA 70128

504/246-8438
504/246-7937

DATA CAPTURE 5.0, **DATA CAPTURE //e**, and **DATA CAPTURE ///** are trademarks of Southeastern Software. Apple][, Apple][Plus, Apple //e, and Apple /// are trademarks of Apple Computer Corporation.

Listing 2

```
10  DIM A%(20),B%(20)
24  D$ =  CHR$ (4)
25  PRINT D$(4);"BRUN
    TURTLESOFT"
100 INPUT "ORDER 1-6 ? ";O
110 IF O < 1 OR O > 6
    THEN  END
120 & "DRAW"
121 & "FS"
130 D = 2
131 IF O <  = 1 THEN 135
132 FOR N = 2 TO O:D =
    D + D: NEXT
135 D = D - 1
140 S =  INT (189 / D)
160 & "PU"
161 & "MT"95, - 95:
162 & "PD":
163 & "TT"90:
164 & "HT"
170 S% = 0
171 I = 0
172 & "HILBERT"
180 GET A$
181 PRINT
182 & "ND"
183 GOTO 100
200 TO "HILBERT"
210 S% = S% + 1
211 A%(S%) = A
212 B%(S%) = B
220 IF I <  > 0 THEN 230
221 & "LT"180
222 GOTO 260
230 IF I < 0 THEN 240
231 A = 90
232 B = I - 1
240 IF I > 0 THEN 250
241 A =  - 90
242 B = I + 1
250 & "HIL1"
251 & "HIL2"
252 & "HIL1"
260 A = A%(S%)
261 B = B%(S%)
262 S% = S% - 1
270 RETURN
300 TO "HIL1"
310 & "LT"A
311 I =  - B
312 & "HILBERT":
313 & "LT"A:
314 RETURN
400 TO "HIL2"
410 & "FD"S
420 I = B
421 & "HILBERT"
430 & "LT" - A
431 & "FD"S:
432 & "LT" - A
440 I = B
441 & "HILBERT"
450 & "FD"S
460 RETURN
```

Patience is a virtue. Computers have lots of it. Humans don't. Humans have lots of computers. Therefore, humans now have lots of patience. Poorly designed software can make humans lose their patience. It is up to software developers to keep the human race virtuous.

The TURTLESOFT version was easily converted to straight APPLESOFT (Listing 3). Where there had been a reference to one of the Turtle functions I coded an equivalent action in APPLESOFT. An example of this can be found in line 410 of Listings 2 and 3. In that line I wanted to make a mark on the screen, or draw a line of the length "S" from where I am currently and in the direction that I previously established. Not much speed was gained, but I did gain a good deal of respect for the relatively high speed of the TURTLESOFT routines written by McBride.

Faster and Faster

I then decided that the majority of the time must be in calculating the vector end points. For those of you who are not conversant in coordinate geometry, the vectors are the line segments. So I translated the APPLESOFT HLINE statements to DRAW statements (Listing 4). I used a very simple shape—it only moved one position down and plotted a point. With this simple shape I only had to set the SCALE to the Hilbert side length, set the rotation factor to point the direction I wanted, and DRAW the shape. I didn't have to keep track of where the shape ended up because that's where I wanted to start from anyway. After establishing a starting position, I didn't have to tell the shape routines where to start the shape. This is the fastest way to use the shape routines, and it improved the speed of the program by a factor of nearly 25%. But I still wasn't happy with the APPLESOFT running ⅓ the speed of the Pascal.

My next step was to bring out the big guns, the TASC compiler that would compile the entire APPLESOFT program. This would surely vanquish the Pascal pretender to the speed throne! Well, it did—but just barely. It was clear at this point that more computational power needed to be brought to bear on the problem.

Again I translated the program—this time to the old stand-by, Integer BASIC (Listing 5). I never did trust those floating points, and it is sometimes difficult to remember that you went out to drain the swamp when you are up to your elbows in alligators. I had translated my program away from the object of my initial desires—that of making the Filler fill a Hilbert space—but speed was my

Listing 3

```
10  DIM A%(20),B%(20)
30  GOTO 100
50  R = R + 4
51  R = R -  INT (R / 4)
    * 4
52  XT = XT + XS(R)
53  YT = YT + YS(R)
54  HPLOT  TO XT,YT:
55  RETURN
100 TEXT
101 INPUT "ORDER 1-7 ? ";O
110 IF O < 1 OR O > 7
    THEN  END
120 HGR
121 HCOLOR= 3
122 POKE 49234,0
130 D = 2
131 IF O <  = 1 THEN 135
132 FOR N = 2 TO O:D =
    D + D: NEXT
135 D = D - 1
140 S =  INT (190 / D)
145 XS(2) = S
146 YS(2) = 0
147 XS(3) = 0
148 YS(3) = S
149 XS(0) =  - S
150 YS(0) = 0
151 XS(1) = 0
152 YS(1) =  - S
153 XT = 45
154 YT = 190
155 HPLOT XT,YT
156 R = 1
160 S% =  - 1
161 I = 0
162 GOSUB 200
180 GET A$
181 PRINT :
182 GOTO 100
200 REM  "Hilbert
210 S% = S% + 1
211 A%(S%) = A
212 B%(S%) = B
220 IF I <  > 0 THEN 230
221 R = R + 2
222 GOTO 260
230 IF I < 0 THEN 240
231 A = 1
232 B = I - 1
240 IF I > 0 THEN 250
241 A = 3
242 B = I + 1
250 GOSUB 300
251 GOSUB 400
252 GOSUB 300
260 A = A%(S%)
261 B = B%(S%)
262 S% = S% - 1
270 RETURN
300 REM   HIL1
310 R = R + A
311 I =  - B
312 GOSUB 200
313 R = R + A
314 RETURN
400 REM    HIL2
410 GOSUB 50
420 I = B
421 GOSUB 200
430 R = R - A
431 GOSUB 50:
432 R = R - A
440 I = B
441 GOSUB 200
450 GOSUB 50
460 RETURN
```

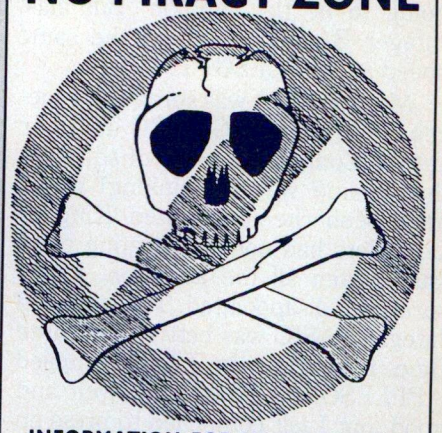
Listing 4

```
 10  DIM A%(20),B%(20)
 19  REM SET UP SHAPE
     TABLE POINTER
 20  POKE 232,0
 21  POKE 233,3
 22  REM NUMBER OF SHAPES
 23  POKE 768,1
 24  POKE 769,0
 25  REM INDEX TO SHAPE #1
 26  POKE 770,4
 27  POKE 771,0
 28  REM SHAPE #1 - DOWN & PLOT
 29  POKE 772,6
 30  POKE 773,0
 32  Q = 16
 33  QH = 32
 34  QM = 64
 35  GOTO 100
100  TEXT
101  INPUT "ORDER 1-7 ? ";O
110  IF O < 1 OR O > 7 THEN  END
120  HGR
121  HCOLOR= 3
122  POKE 49234,0
130  D = 2
131  IF O <  = 1 THEN 135
132  FOR N = 2 TO O:D =
     D + D: NEXT
135  D = D - 1
140  S =  INT (190 / D)
150  HPLOT 235,190
151  SCALE= S
152  R = 16
160  S% =  - 1
161  I = O
162  GOSUB 200
180  GET A$
181  PRINT
182  GOTO 100
200  REM  "Hilbert
210  S% = S% + 1
211  A%(S%) = A
212  B%(S%) = B
220  IF I <  > 0 THEN 230
221  R = R + QH
222  GOTO 260
230  IF I < 0 THEN 240
231  A = Q
232  B = I - 1
240  IF I > 0 THEN 250
241  A =  - Q
242  B = I + 1
250  GOSUB 300
251  GOSUB 400
252  GOSUB 300
260  A = A%(S%)
261  B = B%(S%)
262  S% = S% - 1
270  RETURN
300  REM   HIL1
310  R = R + A
311  I =  - B
312  GOSUB 200
313  R = R + A
314  RETURN
400  REM    HIL2
410  ROT= R
411  DRAW 1
420  I = B
421  GOSUB 200
430  R = R - A + QM
431  R = R -  INT (R / QM) * QM
432  ROT= R
433  DRAW 1
434  R = R - A + QM
440  I = B
441  GOSUB 200
450  ROT= R
451  DRAW 1
460  RETURN
```

Listing 5

```
  5 POKE 74,0
  6 POKE 75,12
  7 XO=YO=COLR=SHAPE=
    ROT=SCALE
  8 REM  LOMEM:3072
 10 DIM AZ(20),BZ(20)
 11 DIM XS(3),YS(3)
 25 PRINT "BLOAD HIRES"
 26 INIT=2809:LINE=2836
 30 GOTO 100
 50 R=(R+4) MOD 4
 51 XO=XO+XS(R)
 52 YO=YO+YS(R)
 53 CALL LINE
 54 RETURN
100 TEXT
101 INPUT "ORDER 1-7 ",O
110 IF O<1 OR O>7 THEN END
120 CALL 2048
121 COLR=255
122 POKE -16302,0
130 D=2
131 IF O=1 THEN 135
132 FOR N=2 TO O
133 D=D+D
134 NEXT N
135 D=D-1
140 S=(190/D)
145 XS(2)=S
146 YS(2)=0
147 XS(3)=0
148 YS(3)=S
149 XS(0)=-S
150 YS(0)=0
151 XS(1)=0
152 YS(1)=-S
153 XO=45
154 YO=190
155 R=1
156 CALL INIT
160 SZ=-1
161 I=O
162 GOSUB 200
180 INPUT A
181 GOTO 100
200 REM   "Hilbert
210 SZ=SZ+1
211 AZ(SZ)=A
212 BZ(SZ)=B
220 IF I#0 THEN 230
221 R=R+2
222 GOTO 260
230 IF I<0 THEN 240
231 A=1
232 B=I-1
240 IF I>0 THEN 250
241 A=3
242 B=I+1
250 GOSUB 300
251 GOSUB 400
252 GOSUB 300
260 A=AZ(SZ)
261 B=BZ(SZ)
262 SZ=SZ-1
270 RETURN
300 REM    HIL1
310 R=R+A
311 I=-B
312 GOSUB 200
313 R=R+A
314 RETURN
400 REM     HIL2
410 GOSUB 50
420 I=B
421 GOSUB 200
430 R=R-A
431 GOSUB 50
432 R=R-A
440 I=B
441 GOSUB 200
450 GOSUB 50
460 RETURN
```


NO PIRACY ZONE



INFORMATION FOR HONEST USERS

Thousands of Apple users have already joined us.

Hardcore COMPUTIST

What you can't get anywhere else.

- Techniques to unlock software
- Unlocking tutorials for beginners
- Tips on program modification
- How to modify DOS
- Game secrets PLUS Advanced Playing Techniques (APT)-how to get those extra ships and weapons
- In-depth product reviews
- Straight answers to your questions

If you're a vigorous Apple computist,

YOU CAN'T AFFORD TO BE WITHOUT US ANY LONGER!

Annual Subscription Rates
Please check one of the following:

- ☐ U.S. $25
- ☐ Canada, APO/FPO, 1st Class $34
- ☐ Mexico $39
- ☐ Foreign airmail $60
- ☐ Foreign surface mail $40

H1

☐ Yes, start my subscription now.

Name _____ Address _____ City _____ State _____ Zip _____ Country _____ Visa/MC _____ Exp _____ Signature _____

Send to: Hardcore COMPUTIST
Subscription Department
P.O. Box 44549H
Tacoma, WA 98444

Circle 9 on Reader Service Card

master now and Integer BASIC was going to be the solution. But alas, Integer BASIC ran about the same speed as APPLESOFT.

My next effort was sure to be a success—using the GALFO Integer BASIC Compiler, a piece of software that (to my way of thinking) would have been the undisputed King of Compilers had not that floating point stuff gotten all the attention. I was surely not disappointed. The compiled Integer BASIC was better than three times faster than the TASC-compiled APPLESOFT. That is all well and good, but I felt compelled to press on toward the ultimate solution—hand-coded assembly language.

Going for the Gold

Now my course was clear. I would not sleep until I knew if the IBC would do as well as I could, programming at the machine's own level. "No sweat", I thought, "I'll write that in 10 or 15 minutes." Well, I used up 15 minutes just trying to find the address of COUT, the program in the monitor that puts a character on the screen.

Listing 6

```
:ASM
                        1   HGR      EQU  $F3E2     CLEAR THE HIRES SCREEN
                        2   HPOSN    EQU  $F411     LOCATE THE HIRES CURSOR
                        3   S        EQU  $00E7     SAVE THE SCALE LOCATION
                        4   SETCOL   EQU  $F6EC     SET HIRES COLOR X<8
                        5   DRAW     EQU  $F601     DRAW SHAPE (Y,X) AT ROT (A)
                        6   RESET    EQU  $03D0     WHERE TO GO WHEN COMPLETE
                        7   RDKEY    EQU  $FD0C     GET ONE CHARACTER FROM THE KEYBOARD
                        8   COUT     EQU  $FDED     PUT ONE CHARACTER ON THE SCREEN
                        9   TXTSET   EQU  $C051     THE TEXT FLIP-FLOP
                       10   D        EQU  $0000
                       11   *
8000: AD 51 C0         12   BEGIN    LDA  TXTSET    100 TEXT
8003: A2 00            13            LDX  #$00          INPUT "ORDER 1-7 ?"
8005: BD 06 81         14   SCRN     LDA  DUES,X         * GET THE NEXT CHARACTER
8008: F0 07            15            BEQ  EOM            * DONE? --> END OF MESSAGE
800A: E8               16            INX
800B: 20 ED FD         17            JSR  COUT           * NO - PUT IT ON THE SCREEN
800E: 18               18            CLC
800F: 90 F4            19            BCC  SCRN           * GO GET THE NEXT CHARACTER
                       20   *
8011: 20 0C FD         21   EOM      JSR  RDKEY     GET A$
8014: 8D 18 81         22            STA  I
8017: 20 ED FD         23            JSR  COUT      PRINT A$;
801A: AD 18 81         24            LDA  I
801D: 38               25            SEC
801E: E9 B0            26            SBC  #"0"      O = ASC(A$) - ASC("0")
8020: F0 06            27            BEQ  DONE      *
8022: 90 04            28            BCC  DONE      IF O < 1 THEN GO TO DONE
8024: C9 08            29            CMP  #$08      IF O <=7 THEN GO TO OK
8026: 90 03            30            BCC  OK         *
8028: 4C D0 03         31   DONE     JMP  RESET     STOP
                       32   *
802B: 8D 14 81         33   OK       STA  ORDER     OK  ORDER = O
802E: AA               34            TAX  *             X = ORDER
802F: A9 02            35            LDA  #$02      130 D = 2
8031: CA               36   FOR      DEX  *             X = X - 1
8032: F0 03            37            BEQ  SIZE          IF X=0 THEN GO TO SIZE
8034: 0A               38            ASL            ;   D = D + D
8035: D0 FA            39            BNE  FOR           GO TO FOR
                       40   *
8037: AA               41   SIZE     TAX  *         135 X = A
8038: CA               42            DEX  *             X = X - 1
8039: 86 00            43            STX  D             D = X
                       44   *
803B: A9 00            45            LDA  #$00      140 A = 0
803D: AA               46            TAX  *             X = A
803E: 18               47            CLC  *             REM ADD (D) TO THE SUM
803F: C9 BE            48   DIV      CMP  #190          REM SUCH THAT SUM < 190
8041: B0 05            49            BCS  SETS          IF A > 190 THEN GO TO SETS
8043: E8               50            INX  *             X = X + 1
8044: 65 00            51            ADC  D             A = A + D
8046: 90 F7            52            BCC  DIV            GO TO RV
8048: CA               53   SETS     DEX  *             X = X - 1
8049: 86 E7            54            STX  S             SCALE = S
                       55   *
804B: 20 E2 F3         56            JSR  HGR           HGR
804E: AD 52 C0         57            LDA  $C052         A=PEEK(49234):REM FULL SCREEN
8051: A0 00            58            LDY  #$00          REM MSB OF X
8053: A2 2D            59            LDX  #45           REM LSB OF X
8055: A9 BE            60            LDA  #190          REM ALL OF Y
8057: 20 11 F4         61            JSR  HPOSN         HPLOT 190,45
                       62   *
805A: A2 20            63            LDX  #32           REM THE ROTATION ANGLE
805C: 8E 17 81         64            STX  ROT           ROT = 32
805F: A2 07            65            LDX  #$07          REM THE HIRES COLOR
8061: 20 EC F6         66            JSR  SETCOL        HCOLOR = 7
                       67   *
8064: AE 14 81         68            LDX  ORDER     160 X = ORDER
8067: 20 70 80         69            JSR  HILBERT       GOSUB 200
                       70   *
806A: 20 0C FD         71            JSR  RDKEY     180 GET A$
806D: 18               72            CLC
806E: 90 90            73            BCC  BEGIN         GO TO BEGIN
                       74   *
8070: 8E 18 81         75   HILBERT  STX  I         200 I = X
8073: AD 19 81         76            LDA  A             A%(S%) = A
8076: 48               77            PHA  *             *
8077: AD 1A 81         78            LDA  B             B%(S%) = B
807A: 48               79            PHA  *             *
807B: 8A               80            TXA  *         220 A = X
807C: D0 0C            81            BNE  L230          IF X<>0 THEN GO TO L230
807E: 18               82            CLC            ;
807F: AD 17 81         83            LDA  ROT           REM 32 = 180 DEGREES
8082: 69 20            84            ADC  #32           R = R + 32
8084: 8D 17 81         85            STA  ROT           *
8087: 18               86            CLC
8088: 90 1F            87            BCC  L260          GO TO L260
                       88   *
808A: 30 0B            89   L230     BMI  L240      230 IF X<0 THEN GO TO L240
808C: CA               90            DEX  *             X = X - 1
808D: 8E 1A 81         91            STX  B             B = X
8090: A9 10            92            LDA  #16           A = 16  (A=1)
8092: 8D 19 81         93            STA  A             *
8095: 10 09            94            BPL  L250          GO TO L250
                       95   *
```


NUMBER COMMANDER
Frustration Minimizer from
ComputerEase, Inc.
1312 W. Cedar St., Appleton, WI 54914

A whole System for Analyses in Natural Sciences and Engineering
Extreme Ease of use – No Computereze
Versatile, Flexible, Tolerant, Safe

Keyboard your data Easily and only ONCE, Analyze ("Play"), Plot, Tabulate, Report.

- **Multiple Regression: linear, logarithmic, exponential, power functions, free form.**
- **122 x 120 Resolution Plots on regular printers:**
  - **"nice" automatic scaling, fast.**
  - **plot variables, ratios, products**
- **Curve Fitting: logarithmic, exponential, power functions, combinations.**
- **Tables of Data, Sums, Averages, Standard Deviations, Skewness, Kurtosis, etc.**
- **Built-in Scientific Calculator.**
- **Type/Print Notes coordinated with Data.**
- **Two COPYABLE Disks. Manual 70 pages.**

Read more in Call-A.P.P.L.E. April -84, p.79-80.

Requires: Apple//e, Apple II Plus-48k with Apple or Smarterm 80-col. board and two disk drives. Compatible Printers: 132 Ch/line, half line feeds, e.g.: Centronics 737, Apple DMP, NEC PC8023A, Spinwriter, EPSON MX/FX-80. Phone: 414/739-7751.

MAIL ORDER FORM TO: ComputerEase, Inc.,
1312 W. Cedar St., Appleton, WI 54914, 414/739-7751
Please send **NUMBER COMMANDER-™** with 2 Disks, Manual and tough Storage Box to:

Name: _____

Address: _____

City: _____ State: _____ ZIP: _____

Terms of Sale: Price: $125:-, Wisc. res. add $6.25 tax. Check one:

__ Master Card, __ VISA, __ Money Order – order shipped immediately

__ Personal Check – allow time for bank clearance

For Credit Card Orders please fill out:

Card no.: _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Expiration Date: _____

Signature: _____

Circle 47 on Reader Service Card

```
8097: E8        96   L240    INX  *        240 X = X + 1
8098: 8E 1A 81  97            STX  B           B = X
809B: A9 30     98            LDA  #48         A = 48  (A=-1)
809D: 8D 19 81  99            STA  A           *
                100  *
80A0: 20 B2 80  101  L250     JSR  HIL1       250 GOSUB 300
80A3: 20 D1 80  102           JSR  HIL2           GOSUB 400
80A6: 20 B2 80  103           JSR  HIL1           GOSUB 300
                104  *
80A9: 68        105  L260     PLA  *          260 B = B%(S%)
80AA: 8D 1A 81  106           STA  B           *
80AD: 68        107           PLA  *           A = A%(S%)
80AE: 8D 19 81  108           STA  A           *
80B1: 60        109           RTS  *           RETURN
                110  *
80B2: 18        111  HIL1     CLC  *          300 R = R + A
80B3: AD 17 81  112           LDA  ROT         *
80B6: 6D 19 81  113           ADC  A           *
80B9: 8D 17 81  114           STA  ROT         *
80BC: 38        115           SEC
80BD: A9 00     116           LDA  #$00        I = -B
80BF: ED 1A 81  117           SBC  B           *
80C2: AA        118           TAX  *           *
80C3: 20 70 80  119           JSR  HILBERT     GOSUB 200
80C6: 18        120           CLC
80C7: AD 17 81  121           LDA  ROT         R = R + A
80CA: 6D 19 81  122           ADC  A           *
80CD: 8D 17 81  123           STA  ROT         *
80D0: 60        124           RTS  *           RETURN
                125  *
80D1: 20 FB 80  126  HIL2     JSR  FORWARD    410 GOSUB 50
80D4: AE 1A 81  127           LDX  B          420 I = B
80D7: 20 70 80  128           JSR  HILBERT        GOSUB 200
                129  *
80DA: 38        130           SEC  *          430 R = R - A
80DB: AD 17 81  131           LDA  ROT         *
80DE: ED 19 81  132           SBC  A           *
80E1: 8D 17 81  133           STA  ROT         *
80E4: 20 FB 80  134           JSR  FORWARD     GOSUB 50
80E7: 38        135           SEC  *           R = R - A
80E8: AD 17 81  136           LDA  ROT         *
80EB: ED 19 81  137           SBC  A           *
80EE: 8D 17 81  138           STA  ROT         *
                139  *
80F1: AE 1A 81  140           LDX  B          440 I = B
80F4: 20 70 80  141           JSR  HILBERT        GOSUB 200
                142  *
80F7: 20 FB 80  143           JSR  FORWARD        GOSUB 50
80FA: 60        144           RTS  *          460 RETURN
                145  *
80FB: A0 81     146  FORWARD  LDY  #>SHAPE    GET MSB OF SHAPES ADDRESS
80FD: A2 15     147           LDX  #<SHAPE    GET LSB OF SHAPES ADDRESS
80FF: AD 17 81  148           LDA  ROT        GET THE ROTATION VALUE
8102: 20 01 F6  149           JSR  DRAW       MAKE A MARK ON THE SCREEN
8105: 60        150           RTS
8106: 8D        151  QUES     HEX  8D         RETURN
8107: CF D2 C4
810A: C5 D2 A0
810D: B1 AD B7
8110: A0 BF A0  152           ASC  "ORDER 1-7 ? "
8113: 00        153           HEX  00         END OF MESSAGE
8114: 00        154  ORDER    HEX  00         HOW COMPLEX A HILBERT
8115: 06 00     155  SHAPE    HEX  0600       ONE SIDE OF A HILBERT
8117: 00        156  ROT      HEX  00         ROTATION ANGLE 64 = 360 DEGREES
8118: 00        157  I        HEX  00         TEMP STORAGE
8119: 00        158  A        HEX  00         *
811A: 00        159  B        HEX  00         *

--END ASSEMBLY--

ERRORS: 0

283 BYTES

SYMBOL TABLE - ALPHABETICAL ORDER:

A        =$8119    B        =$811A    BEGIN    =$8000    COUT     =$FDED
D        =$00      DIV      =$803F    DONE     =$8028    DRAW     =$F601
EOM      =$8011    FOR      =$8031    FORWARD  =$80FB    HGR      =$F3E2
HIL1     =$80B2    HIL2     =$80D1    HILBERT  =$8070    HPOSN    =$F411
I        =$8118    L230     =$808A    L240     =$8097    L250     =$80A0
L260     =$80A9    OK       =$802B    ORDER    =$8114    QUES     =$8106
RDKEY    =$FD0C    RESET    =$03D0    ROT      =$8117    S        =$E7
SCRN     =$8005    SETCOL   =$F6EC    SETS     =$8048    SHAPE    =$8115
SIZE     =$8037    TXTSET   =$C051

SYMBOL TABLE - NUMERICAL ORDER:

D        =$00      S        =$E7      RESET    =$03D0    BEGIN    =$8000
SCRN     =$8005    EOM      =$8011    DONE     =$8028    OK       =$802B
FOR      =$8031    SIZE     =$8037    DIV      =$803F    SETS     =$8048
HILBERT  =$8070    L230     =$808A    L240     =$8097    L250     =$80A0
L260     =$80A9    HIL1     =$80B2    HIL2     =$80D1    FORWARD  =$80FB
QUES     =$8106    ORDER    =$8114    SHAPE    =$8115    ROT      =$8117
I        =$8118    A        =$8119    B        =$811A    TXTSET   =$C051
HGR      =$F3E2    HPOSN    =$F411    DRAW     =$F601    SETCOL   =$F6EC
RDKEY    =$FD0C    COUT     =$FDED
```

The next few hours were spent remembering how to do a divide without doing a divide, since divide and multiply are not part of the 6502 instruction set. As you may notice in lines 31 thru 35 of the Assembly Language program (Listing 6), I was able to use the same trick that had saved Pascal from the loss of precision in calculating two to a power. Normally, statement 130 in the BASIC versions would have been coded as "130 D=2↑0," which would have been interpreted as EXP(LOG(2)*0)—taking the log of 2, multiplying that by 0, and then raising e to that power. Both LOG and EXP are calculated by a polynomial, so that would mean a bunch of multiplies and adds. As a programmer, you must make the decision between speed, readability, and easy-to-write-and--maintain code. The way statements 130–135 are coded is faster, but it's only run once so it doesn't matter much in BASIC. In assembly language it's the only reasonable way to do it. The comments portion of the Assembly Language listing has a BASIC-like structure so you can compare the BASIC and Assembly Language programs more easily.

FREE?DISKETTES

SAVE MONEY! Apple II+/e users can use the diskette flip side, if another "write enable" notch is correctly made.

The DISK-NOTCHER by QUORUM quickly SOLVES that PROBLEM. It's like FREE DISKETTES!

- Stainless Steel Guide
- Easy Leverage Handle
- Clippings Catcher
- Square Notch Cut
- Black Finish
- Get THE BEST!

Certifix™

BE SAFE! Your 'FREE' disk is CERTIFIED 100% ERROR FREE with CERTIFIX by QUORUM. It 'LOCKS OUT' DISK FLAWS and lets you use the rest. Displays status report & saves it to disk. Next, CERTIFIX automatically formats then offers to initialize with genuine Apple DOS 3.3 too. Great for testing economy disks. CERTIFY, FIX & INITIALIZE every disk with CERTIFIX!

100% Money Back Satisfaction Guarantee!

DISK NOTCHER is $14.95
CERTIFIX™ is just $24.95
ONLY $29.95 for BOTH!
Add $1.50 s/h • CA add 6½ % tax

QUORUM INTERNATIONAL, Unltd.
INDUSTRIAL STATION P.O. BOX 2134-CA
OAKLAND, CA 94614

Circle 111 on Reader Service Card

Table 1

Times Measured in Seconds

System	Language	Complexity 1	2	3	4	5	6	7
Apple	Assembler Language	.33	.38	.47	.68	1.24	2.70	6.43
Apple	IBC Integer (Shapes)	.33	.42	.60	1.42	4.52	15.95	59.38
Apple	IBC Integer Basic	.33	.42	.60	1.64	5.20	19.08	72.51
Apple	Compiler + (Shapes)	.33	.56	1.10	3.30	11.05	42.00	162.38
Apple	TASC Compiled	.33	.54	1.42	5.13	16.92	66.01	261.20
Apple	Apple Pascal	.94	1.66	2.29	5.74	20.26	78.50	314.93
Mac	MicroSoft Basic	.19	.62	2.29	8.92	36.13	141.45	565.35
Mac	Pascal (Think Tech.)	.23	.63	2.57	9.92	39.09	155.85	623.26
Sperry	GW-Basic	.56	1.30	3.91	13.46	51.33	201.61	800.00
Apple	AppleSoft (Shapes)	.76	1.64	4.74	16.72	64.40	258.97	1035.88
IBM PC	MicroSoft Basic	.86	1.66	5.03	18.02	69.73	275.83	1103.32
Apple	AppleSoft	.76	1.87	5.73	21.25	83.49	332.26	1329.04
Apple	Integer Basic	.77	1.84	5.87	23.23	92.46	368.90	1475.60
Apple	TurtleSoft	.82	2.01	8.16	30.40	120.89	480.25	1921.00

Relative Ratios of Times

System	Language	Complexity 1	2	3	4	5	6	7
Apple	Assembler Language	1	1	1	1	1	1	1
Apple	IBC Integer (Shapes)	1	1	1	2	4	6	9
Apple	IBC Integer Basic	1	1	1	2	4	7	11
Apple	Compiler + (Shapes)	1	1	2	5	9	16	25
Apple	TASC Compiled	1	1	3	8	14	24	41
Apple	Apple Pascal	3	4	5	8	16	29	49
Mac	MicroSoft Basic	1	2	5	13	29	52	88
Mac	Pascal (Think Tech.)	1	2	5	15	32	58	97
Sperry	GW-Basic	2	3	8	20	41	75	124
Apple	AppleSoft (Shapes)	2	4	10	25	52	96	161
IBM PC	MicroSoft Basic	3	4	11	27	56	102	172
Apple	AppleSoft	2	5	12	31	67	123	207
Apple	Integer Basic	2	5	12	34	75	137	229
Apple	TurtleSoft	2	5	17	45	97	178	299

Orders of Hilbert

Complexity	Number of Lines	Line Length	Points (Pictles Changed)
1	3	190	570
2	15	63	945
3	63	27	1701
4	255	12	3060
5	1024	6	6144
6	4096	3	12285
7	16383	1	16383

A HILBERT space filling curve of Order 1 has three adjoining curves, each at right angles to the last, thus forming a box open on one side. On the Apple II hi-res screen, this allows each line to be 190 points long. A HILBERT space filling curve of Order 6 has 4096 lines of length 3, each at a right angle to the last.

A Happy Ending

Although I never quite got around to having the Filler fill in a Hilbert curve (nor did I drain the swamp), my struggles with all those language alligators produced an unintended but valuable result. I finally realized that all this work amounted to a new form of graphics-oriented benchmark for comparing the speeds of different interpreters and compilers. These comparisons are summarized in Table 1. In addition to the Apple II programs mentioned above, this table includes the outcomes of BASIC versions of HILBERT run on the Apple Macintosh (Microsoft BASIC), the IBM PC (Microsoft), and other selected systems and compilers.

MORE INFORMATION about the products and services in this issue FREE!

Just circle the number and fill out the card. (Note: card expires three months following month of publication.)

Act Now! Join! A.P.P.L.E.

Apple **P**ugetSound **P**rogram **L**ibrary **E**xchange

The **World's** Largest Apple Computer User Group

Be a part of the over 20,000 A.P.P.L.E. members and you'll receive:

- 12 issues of Call–A.P.P.L.E. magazine. The nation's premier Apple computer publication.
- Substantially reduced prices on sophisticated software, hardware, publications, and services.
- Access to the A.P.P.L.E. closed user group on the **SOURCE.** Excellent for information exchange and program downloading.
- **Call —A.P.P.L.E.** Hotline.

ALL THESE BENEFITS AND MORE FOR ONLY $51!

Detach and mail the Business Reply Application Form. →

READER SERVICE CARD

Free Information on the Items circled

Card expires September 30, 1984

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150

SURVEY

What one computer do you currently use most?

A ☐ *Apple II*
B ☐ *None*
C ☐ *Apple IIe*
D ☐ *Other (specify)*

What computer (if any) would you likely purchase in the near future?

K ☐ *Apple IIe*
L ☐ *Lisa*
M ☐ *Macintosh*
N ☐ *Apple III*
O ☐ *IBM*
P ☐ *Apple IIc*

What hardware product are you most likely to buy in the next 4 months?

E ☐ *Hard Disk*
F ☐ *5 megabyte*
G ☐ *10 megabyte*
H ☐ *20 megabyte*
I ☐ *8086 board for your Apple (run IBM P/C software)*
J ☐ *Apple keyboard (increase capability)*

Name _____

Company _____

Address _____

City _____

State _____ Zip _____

(7/84)

Call —A.P.P.L.E. Reader's Choice

Help us help you. By filling out and returning this card, marked with your preferences, you can guide us in selecting material tailored to your needs. Just place the numbers from 1 to 10, in order of preference, in the appropriate boxes. Please do not choose more than 10 features. A.P.P.L.E. members who include their membership number will have their names placed in a drawing for $100 credit toward any A.P.P.L.E. product. Drawing will be held September 18, 1984 The top scoring author will also receive a $100 cash bonus.

☐ **Computer Languages** Jonathan V. Post

☐ **Bringing up FORTH** Andrew Bernat

☐ **Hilbert and the Alligators** Don Williams

☐ **Microsoft BASIC on the Macintosh** Tom Sethre

☐ **LISP on the Apple II** Vince Giambalvo

☐ **Book Review: Apple Assembly Language** Lee Webster

☐ **Speaking of Tongues: A Review of Foreign Language Learning Software** Kaiulani Schuler

☐ **Logo vs. Logo** Jeff Sandys

☐ **Northern Spy: Managing With Ampermanager** Rick Sutcliffe

☐ **Something Over 32 Bits** Dave Redhed

☐ **Running APPLESOFT Programs With ProDOS** Richard D. Norling

☐ **A.P.P.L.E. Doctor** Bruce F. Field

☐ **The Inner Circuit** Don Elman

☐ **Write—A.P.P.L.E.**

☐ **Member Billboard** Don Elman

☐ **The Marketplace**

Name _____ Address _____ Memb. # _____

A.P.P.L.E. employees ineligible for prizes. Void where prohibited by law. (7/84)

☐ Please send me an A.P.P.L.E. Apple-cation and information.

☐ $51.00 payment enclosed.

☐ Mastercharge/Visa

$25.00 First time fee
26.00 First year dues
$51.00 Total

MasterCard # _____ Exp. Date _____

VISA # _____ Exp. Date _____

Name _____

Address _____

City _____ State _____

Zip _____ Phone _____

Apple PugetSound Program Library Exchange
The WORLD'S largest Apple Computer user group
A.P.P.L.E., 21246 68th Ave. S., Kent, WA 98032
(206) 872-2245

First Class mail for **Call —A.P.P.L.E.** add $15. Otherwise third class mailed free as part of membership in U.S.A.

Additional Airmail postage required for foreign memberships and shipments.

Additional membership postage:
- Canada, West Indies, Central America, Columbia, Venezuela and Mexico — Add $15.00 ($66.00 total)
- All S. American countries (except above), Europe, and N. Africa — Add $21.00 ($72.00 total)
- Asia, Mid East, S. Africa, Australia and Pacific Islands — Add $27.00 ($78.00 total)

All orders must be prepaid in U.S. funds only.

(7/84)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 217 CLINTON, IA

POSTAGE WILL BE PAID BY ADDRESSEE

Call-A.P.P.L.E.
P.O. Box 1993
Clinton, IA 52735

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 32 KENT, WA

POSTAGE WILL BE PAID BY ADDRESSEE

Call-A.P.P.L.E.
21246 68th Ave. S.
Kent, WA 98032

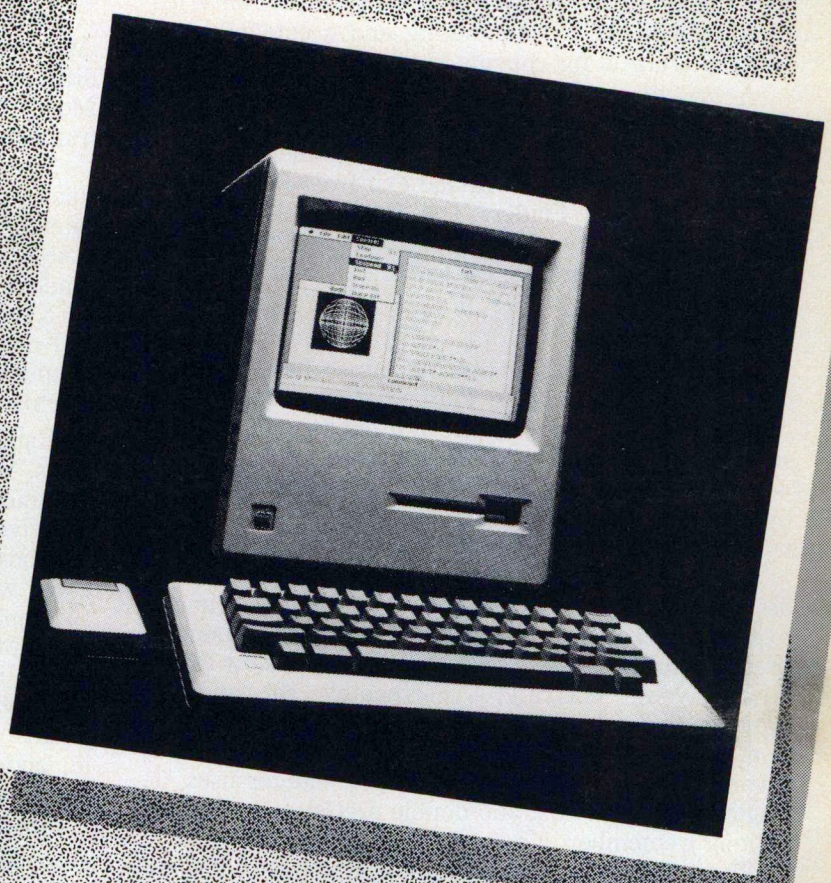
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 32 KENT, WA

POSTAGE WILL BE PAID BY ADDRESSEE

A.P.P.L.E.
21246 68th Ave. S.
Kent, WA 98032

Microsoft BASIC on the Macintosh



Tom Sethre

AS A LONG-TIME Apple II owner, I am accustomed to a computer with ROM-resident BASIC. Wanting to program the Macintosh a bit, I purchased Microsoft BASIC the week my Mac arrived. After years of Applesoft, this BASIC seems a mixture of familiar constructs and powerful new commands. However, its implementation in the Macintosh environment is critically flawed, leaving me somewhat baffled as to who is meant to use it.

Booting

Microsoft BASIC is started like any other Macintosh application program—by either opening the icon that represents the application, or by opening the icon of a document (BASIC program) created by the application. Either choice loads the Microsoft BASIC Interpreter into the Macintosh's RAM. If a program was specified, it is then loaded and run; if not, the program workspace is cleared for entry of a new program.

Windows

Interaction with the interpreter occurs through the menu bar across the top of the screen and three types of windows: the *Command* window, the *Output* window, and *List* windows. When the Command window is present, the interpreter will accept keyboard input. This window disappears whenever a program is running. When you enter a command or program line, it first appears in the Command window where it can be edited. Once the [RETURN] key is pressed, the completed line is echoed on the Output window, and if it was a program line, it may also appear in a List window. The Output window is similar to what appears on our Apple II's — it scrolls up a line for each keyboard entry and displays program lines and output. Its size and screen location may be changed using the mouse, but its contents cannot be directly altered.

There can be up to three active List windows, each displaying a different portion of the program in memory. You may change their size, shape, and screen location, or "close" them to make them go away. (List windows automatically "drop behind" the Output window when a program is running.) List windows also have scroll bars along the side and bottom, so any portion of the program can be quickly displayed, and long lines can be scanned. Having multiple listing windows to look into long programs is a particular boost to program development.

Editing Programs

Editing of program lines is accomplished by pointing to the desired line in a List window and clicking the mouse. The selected line then appears in the Command window, where it may be changed using any of the standard Macintosh text editing techniques. When done, press RETURN to clear the Command window. The new line will scroll up the Output window, and the List window will be updated to reflect the change.

The process of entering programs is a lot easier than with the old Apple II, even with GPLE. Additional features include AUTO numbering and a cut-and-paste clipboard, useful for copying blocks of complex code.

Across the top of the screen is the standard Menu bar. In addition to the permanent Apple menu, there are File, Edit and Control menus. The File menu lets you load a program file, save it, and exit from BASIC back to the Macintosh Finder. The Edit menu gives you access to the clipboard. The Control menu lets you start (run), suspend, continue, list and trace programs.

Variables

Microsoft BASIC on the Macintosh has a decimal math package with up to fourteen digits of precision. Numeric variables may be one of three degrees of precision. Ingeger is our old friend with the -32768 to +32767 limit, internally stored as 16 bits (two old-style bytes). Next comes single precision (which takes up four bytes), followed by double precision (eight bytes).

Unless you specify otherwise, numeric variables are assumed to be double precision. What's more, numeric expressions are always evaluated with the maximum precision found in the expression. This can slow things down if you're not careful. Fortunately, the language provides two ways to specify (and limit) variable precision. One way is to append a type character at the end of a variable's name. Strings use the $, integer uses %, single precision uses !, and double precision uses either nothing (remember, it's the default) or #.

The other way to set a variable's type is to declare that any variable which has no explicit type suffix character and that begins with a letter in the range ". . . - . . ." shall be of a certain type. Say, for example, that you decide to reserve all variable names starting A through F to be string, those starting G through L to be integer, and M through R to be single precision. You declare this with the following statements:

```
110 DEFSTR A-F
120 DEFINT G-L
130 DEFSNG M-R
```

Spaces Count

At this point, Applesoft users start to have nightmares, because keeping track of variable names is always such a pain. No more with this BASIC, since variable names can be up to *forty* characters long (alphanumeric, first alpha). Because the parser scans for blanks to separate words, a variable can even contain part of a reserved word without trouble. (Applesoft scans the line character by character without regard to blanks, so we had to be careful not to incorporate its reserved words in variable names.)

The other side of this coin is the fact that *spaces matter*! No longer can you merrily enter a line without ever touching the space bar. In that case, Microsoft BASIC might think you are using a long variable name. For example, GOSUB2000 will trigger a syntax error at run-time because the parser will consider it a variable name rather than a command. Fortunately, if you forget to put spaces between words, editing is fairly painless.

Data File Management

Most of the language's commands are familiar to Applesoft users, but there are a few that have slightly different characteristics, and a lot of powerful new ones. Most of these new commands relate to the built-in (rather than DOS 3.3 "tacked-on") file I/O.

The new Input/Output commands are very powerful. Files are opened and assigned numbers, and subsequent references to the files are by the numbers. Microsoft BASIC on the Macintosh recognizes sequential files and random files. Sequential files are much like DOS 3.3 TEXT files, but the access is much cleaner thanks to an end-of-file function. They may be accessed as Input, Output or Append, which allows you to add to the end of an existing sequential file.

Random files are also similar to the Apple II counterparts, with the record length declared in the OPEN statement used to calculate the displacement into the file of a given record. However, reading and writing records to random files is accomplished in a somewhat roundabout way. First, the file's buffer is "fielded." This is accomplished by defining the length and string data name of each field in the record. As each random file record is read (through a special GET statement), the string variables named in the "fielding" statements are filled in. More to the point, the file's record buffer is filled in, and the preceeding FIELD statements define the names assigned to portions of the record buffer.

Notice that only string fields are allowed. In order to store and retrieve numerics, a special set of functions are present that convert numerics to strings. However, the resulting strings are not expanded, but are the same length as the internally represented number. In other words, each number must be converted to something that the machine will treat like a character string, but the binary contents if the string is no different than the binary image of the original numeric variable. However, strings also have a five-byte overhead in addition to the actual length of the string.

To work with any numeric information you must explicitly convert the "numeric string" fields into true numerics. Let's say you have a field in file number 3's records that holds a salary. You name it SALARY$ in the following FIELD statement:

```
2760 FIELD -3, 10 AS SALARY$
```

This defines ten characters of the file number three's record buffer as the string SALARY$. Field statements may name several fields, and several statements may be used to fully describe a record.

Usually you deal with salary as a single precision variable, so you must next convert it to SAL! using the following command:

```
3240 SAL! = CVS(SALARY$)
```

where CVS means ConVert to Single precision. There are also CVI (integer) and CVD (double) functions, along with the converse set of functions MKS$, MKI$ and MKD$ for preparing numbers for storage in the record buffer.

When you are ready to place a number back on the file, you first make it into a string (MKI$, etc.), and then "set" it into the record buffer field with a LSET (left set) or RSET (right set) command. Once all the fields have been set, you then PUT the record onto the file.

Although this seems cumbersome at first, it does provide the programmer with very precise control of record content and length. My impression is of a very professional set of file I/O processes.

More Niceties

Control structures in the language are enhanced. The familiar GOTO, GOSUB and FOR . . . NEXT are there, along with ON . . . GOTO and ON . . . GOSUB. The conditional IF . . . THEN has been enhanced to a full IF . . . THEN . . . ELSE, and a shorthand IF . . . GOTO has been thrown in. A special treat is the addition of a WHILE . . . WEND structure, controlling a block of instructions while a certain condition holds true.

The mouse has its own set of functions, making it very easy to incorporate its use into programs. Other specifically "Macintosh" elements are the TIME$ and TIMER functions, which take advantage of the standard clock, and over forty pre-defined calls to the Mac's built-in graphics routines. These calls don't use numbers as called addresses, as we are so used to doing: no more CALL -956. Instead, calls have pre-defined names such as MoveTo, TextFace, PaintOval and FillRoundRect.

Is There a Catch?

After all this, you may begin to think that this is a pretty comprehensive language. In many ways it is, but I find that there are also key elements lacking. These relate to the special nature of the Sony 3.5" disk that has become the standard of the Apple 32-bit product line.

In order to change disks on the Macintosh, you have to request that the program eject the current disk. You can't just yank it out as with the old style 5 ¼" floppies. This is not necessarily bad — in fact it helps insure data integrity.

Unfortunately, Microsoft BASIC on the Macintosh does not provide for ejecting the disk. There are no commands to do so from within a program, no special CALLs to the Mac's ROM routines . . . nothing. What's more, the FILE menu up on the menu bar lacks an EJECT, as do the LOAD and SAVE commands. You can't write a program and then SAVE it to a data disk. You are forced to save your program on the same disk that has the interpreter and system files. If you are working on a program on one disk, and want to work on a program on another disk, you must exit completely out of BASIC, change disks, and re-start BASIC using its image from the new disk. A copy of the interpreter must be present on the program disk.

This also means that a program's data files have to reside on the program disk, unless you have one of the hard-to-obtain additional drives. Even with the additional drive, if an incorrect disk is in the drive when Microsoft BASIC is loaded, your programs can't do a thing about it.

Interestingly enough, the language lets you specify the disk volume of a file at open time, and will trigger an error if the volume cannot be found. You can even write error trapping routines to recognize this case. It won't do a whole lot of good though, because you can't clear the offending disk for the operator to insert the correct one.

This strikes me as very strange. Here we have a powerful language, with excellent I/O capabilities carefully implemented, but then we are left without a way to access the storage needed to support the files.

Where Do We Go From Here?

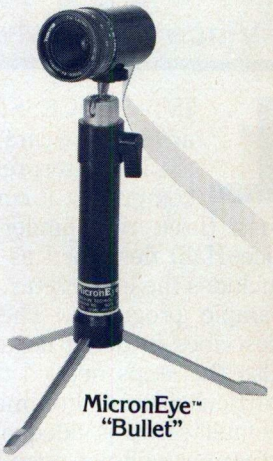
I feel like I've been invited to a wonderful new country ("Macland") and given the keys to a sports car with which I am to explore the countryside. Upon entering the garage, the car's sleek lines, shiny chrome and powerful engine at first blind me to the fact that the vehicle is up on cinder blocks, without wheels.

Just who is this product's intended consumer? It certainly isn't the professional developer (who is expected to use a compiled language anyway). It can't be for the custom business developer either, without access to disk-swapping routines. In fact, any business application seems ruled out, which makes me wonder why they went to all the trouble with the file I/O. Is this package only to be used to teach BASIC? Are the multiple numeric types, the complex PRINT USING options, the well thought-out control structures only there for *dabblers*?

If I sound frustrated, I am. My only hope is that, since I have Version 1.00.00, the gross oversights will be corrected in subsequent releases. In the meantime, I'm waiting for a Macintosh C Compiler.

Now Your Computer Can See!

$295.00*



MicronEye™ "Bullet"

A total imaging system complete and ready for plug-and-go operation with your personal computer.

The MicronEye™ offers selectable resolution modes of 256 x 128 and 128 x 64 with operating speeds up to 15 FPS. An electronic shutter is easily controlled by software or manual functions, and the included sample programs allow you to continuously scan, freeze frame, frame store, frame compare, print and produce pictures in shades of grey from the moment you begin operation.

Only the MicronEye™ uses the revolutionary IS32 OpticRAM™ image sensor for automatic solid state image digitizing, with capability for grey-tone imaging through multiple scans. And with these features, the MicronEye™ is perfectly suited for graphics input, robotics, text and pattern recognition, security, digitizing, automated process control and many other applications.

The MicronEye™ is available with immediate delivery for these computers: Apple II, IBM PC, Commodore 64 and the TRS-80CC (trademarks of Apple Computer Inc., International Business Machines, Commodore Corp., and Tandy Corp. respectively).

Phone for MicronEye™ information on the Macintosh, TI PC and RS232 (trademarks of Apple Computer Inc. and Texas Instruments respectively.)

*(Add $8.00 for shipping and handling [Federal Express Standard Air]; residents of the following states *must add sales tax:* AK, AZ, CA, CO, CT, FL, GA, IA, ID, IL, IN, LA, MA, MD, ME, MI, MN, NC, NE, NJ, NY, OH, PA, SC, TN, TX, UT, VA, VT, WA, WI.)

MICRON TECHNOLOGY, INC.

VISION SYSTEMS
2805 East Columbia Road
Boise, Idaho 83706
(208) 383-4106
TWX 910-970-5973

Circle 66 on Reader Service Card

LISP

on the Apple II

Product:	P-Lisp
Use:	LISP Programming language
For:	Apple II family
Company:	Gnosis 4005 Chestnut Street Philadelphia, PA, 19104
Includes:	Disk with interpreter and sample programs Manual Tutorial book
Price:	$99.95

Vince Giambalvo

LIST-PROCESSING language (LISP) has been around for years on larger computers. P-LISP is a LISP interpreter for the Apple II series of computers. It comes in a package with a manual in a three ring binder, a book (published by Prentice-Hall) described as a tutorial, and a diskette that includes the interpreter, and several files of utilities and sample programs. The manual is well written and describes in detail not only the built-in functions, but also the steps needed to configure the interpreter for particular uses. For many applications, however, the default setup is adequate, and the inexperienced programmer will not even have to look at this part of the manual. The tutorial is designed to teach a novice. It is well written and seems to do its job.

What the Package Contains

The sample programs included with P-LISP give a hint as to the power of the language. Included are ELIZA, a version of the famous psychiatrist program, and HANOI, a version of the childrens game Towers of Hanoi. ELIZA is frustratingly simpleminded, due to the severe memory restrictions (it runs in 48K; more could be done if additional memory were used). HANOI was included to demonstrate the graphics commands in P-LISP (which are almost identical to Applesoft), but it also illustrates the philosopy of LISP programs—given a complicated problem, reduce it to a similar, but simpler problem until the simpler problem is trivial. Also included as utility programs are an editor, a prettyprinter, and a package for disk I/O.

There are not as many built-in functions as might be found in a LISP for a larger machine, but enough to get the job done. For example, most LISP implementations provide—in addition to the standard functions CAR (which gives the first element of a list) and CDR (which deletes the first element of a list)— functions to find the second element (CADR), to delete the first two elements (CDDR), and many other combinations. These extended functions are not present in P-LISP, but there is a function called ARRAY which will find any element of a list. There are several functions designed to take advantage of special features of the Apple, such as high resolution graphics. These are almost the same as Applesoft. The transcendental functions (sin, cos, etc) are provided in a separate file so that no memory is required for them if they are not used. The graphics functions can also be deleted for additional memory.

The DISKIO file contains LISP functions to read and write data to disk, and to store function definitions. The stored functions can be recalled by EXECing the file. As an alternative to storing individual functions, the entire workspace can be saved and reloaded with the SAVE and LOAD commands. These work exactly like the DOS commands for BASIC programs— but they are different. Because they deal with the entire workspace, a save can use 88 sectors. SAVE creates a file with type "S," so other programs won't be tempted to use its files. All the other DOS commands work as usual.

Support for a printer and/or an 80 column card is not explicitly provided, but the PR# commands work. Since P-LISP uses standard Apple I/O, these peripherals function well—with one problem. P-LISP has only one print driver and it is set up to give a nice display on the screen. In particular, when the PRINT function prints a list, it prints a carriage return only at the end. It also prints some extra blanks before right parentheses, and cancels these with backspaces. If your printer does not support backspaces, or will not insert carriage returns, you may have problems.

Editing

There are two ways to edit functions. The first is to use the editor supplied with P-LISP. This is a useful program, but has its deficiencies. It has no provision for moving parentheses, which is unfortunate since forgetting a parenthesis is one of the most common typing errors. The second method is to write the function definition to a textfile, use a text editor to correct it, and then EXEC it back in. This is cumbersome, but it works if your text editor can handle the textfiles created by P-LISP (see previous paragraph). PIE Writer has no problems at all with these files. The editor is almost a standard LISP function editor. It is set up to work well on LISP function definitions, but it does not resemble PIE or WordStar at all. You must read the manual before using it.

It is a good idea to write your complete program to disk occasionally, reinitialize the interpreter, and then EXEC the program back in. This is because LISP remembers all the words you have typed in, including misspellings, tests, etc. Thus memory will get clogged with unneeded junk. The above process will clear it out.

Memory

Memory is divided into three parts. The first is that needed by the Apple itself and DOS, the second the interpreter ($800 - $3FFF). The rest is divided between the workspace, the recursion stack, and any memory that the user wishes to reserve—for graphics, machine language subroutines, etc. If a 16K RAM card is available, the user can tell the interpreter to use it by adjusting the memory map. This is an easy procedure, described in detail in the manual. The one deficiency in the manual is the lack of a description of the internal representation of LISP objects, and instructions on interfacing machine language programs to LISP. This information is provided in a newsletter from Gnosis.

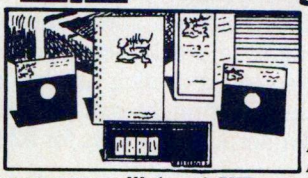
P-LISP is very efficient in its memory usage. But the workspace and stack have at most about 33 ½ K. This is small for LISP, but it is not a disaster. It is possible to run significant application programs.

My interest in LISP lies in computer algebra. Microsoft does market a LISP-based computer algebra system called muMATH, but it requires a Z-80 card. Since my Apple does not have one, I looked for a version of LISP for a standard Apple. Because most of the programs I would need were already running on a mainframe, P-LISP looked like the answer—if a virtual megabyte could be squeezed into 64K. I was very surprised when, with a very few modifications, everything (or at least enough for my application) fit into 48K. While no speed demon, P-LISP seems to run about a tenth the speed of the LISP on the mainframe, which was fast enough for me.

Recommendation

In summary, P-LISP is a credible implementation of LISP which runs well on a standard Apple. With its documentation it is accessible to a beginner, and it has enough power to be useful to the experienced programmer. If you are still undecided about whether or not you want to use LISP, get a copy of the August 1977 issue of *Byte*. It contains several articles on applications of LISP, as well as some basic information.

Triple Your Programming Speed!



GUARANTEED!

APB-102 **$189.00**

Adds mainframe editing power to your Apple's language! Uses **NO** program memory space. Works with **ANY** size program! Hundreds of programming features never before combined in one package! **ALWAYS AVAILABLE** and **NEVER IN THE WAY!** The features of this system are far too extensive to describe here, so call or write for more information, and get a **FREE** quick reference card.

OEM'S PLEASE NOTE: Now you can market your turnkey systems in ROM! Eliminate worries about copy protection. Your program auto-boots from ROM which can even eliminate the need for a disk drive. **Call or write for details.**

Optional Additions Include:

NEW!! APU-2: supeRENUMBER, multiHIDE, multiMERGE, VARIABLE CROSS REFERENCE, AUTONUMBER, NEW programs by Paul Johnson. Far more powerful and mistake proof than other versions...and always at hand. **$35.00**

NEW!! COPY ZAP by Glen Bredon, author of BIG MAC. SMART DISK COPIER, and ZAP utilities written for our system. Also copies 40 track disks! **$39.95**

NEW!! ROM DEVELOPMENT SYSTEM: Program your own firmware! Includes software, documentation and emulation RAM. **$99.00**

12 Bit 16 Channel A/D Convertor

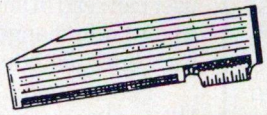
AD16B **$299.95**

★VERY HIGH SPEED
★7 VOLTAGE RANGES
★COMPLETE SOFTWARE

SIGNAL CONDITIONING & DIFFERENTIAL INPUTS AVAILABLE

Professional Prototyping Board

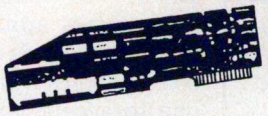
THE BIGGEST—THE BEST



PRO-1 **$29.95**

★IDEAL LAYOUT FOR WIRE WRAPPING
★GOLD PLATED EDGE CONNECTOR
★VERY LARGE—UP TO 52 (16 PIN) SOCKETS

48 Line Parallel I/O



PD48 **$249.95**

★48 LINES IN/OUT (SOFTWARE SELECTABLE)
★6 TIMERS PLUS INTERRUPT
★FULL DOCUMENTATION

APPLE IS A REGISTERED TRADEMARK OF APPLE COMPUTERS INC.

VISA MasterCard **Hollywood Hardware** 6842 VALJEAN AVENUE, VAN NUYS, CA 91406 (818) 989-1204

Circle 48 on Reader Service Card

Northern Spy:

Managing With Ampermanager

Rick Sutcliffe

THIS MONTH I'd like to consider how an "Amper Utility" works and what it can do for you. Then I'll describe some of the features of *Ampermanager* (a program I wrote that is available through A.P.P.L.E.), along with programming suggestions. I'll also discuss how it can be modified to work under ProDOS and, finally, will explain how to create a unique, hybrid DOS 3.3/ProDOS disk which will boot and run programs under either system.

How "&" Works

The Applesoft language contains a command which is just a bare ampersand (&). Whenever the interpreter encounters this command symbol in a running program (or input line, which is essentially the same thing), it does a jump to the machine language location $3F5. DOS places at that location another jump to an RTS (return) command, and control would normally pass back to the program with nothing having happened at all.

If a programmer wishes the & to become a "live" command, it is necessary to change the jump at $3F5 to point to some other location containing machine language code which actually does something. If the programmer is familiar with the way in which BASIC parses statement lines, the & can be followed by other syntax such as &C (for CATALOG), &PRINT USR (for formatted output), or &TYPE (to find out what's in a text file), and so on.

Indeed, a very long list of additions to BASIC can be made up, and these extra commands—once preceeded with an &—can be interpreted and executed by the program which lies at the destination of the jump now in $3F5. Routines suitable for inclusion in such a package can be found in many technical publications about Apple computing, and others can be developed by a skilled programmer.

The Ampermanager Package

The current disk sold by A.P.P.L.E. includes Ampermanager in three identically performing versions—low memory, high memory, and relocatable (R file). This program has sixty utilities, including INPUT anything, PRINT Using, various string substitutions (&MID$(A$,2,3) ="2E"), numerous DOS utilities including a full-fledged disk editor, a TEXT file TYPE, and many other major and minor command additions.

There are three Applesoft driver programs which serve as sophisticated utilities in their own right and as programming suggestions for the use of Ampermanager. The simplest is a program for writing, sorting, and editing short line-oriented text files which are intended to become EXEC routines for automatically controlling an Apple. The user can read in an existing file (or create a new one), change it around, add or subtract lines, and then write it back.

The second Applesoft utility is a DOS editor, the principal function of which is to change some of the DOS 3.3 routines around. Commands and error messages can be edited and some patches can be made—including one to accept lower-case DOS commands in addition to the normal upper-case ones.

The third is primarily intended to shuffle the CATALOG around. With it, you can sort an existing disk catalog, remove deleted files, undelete files which have not been written over, and do some general housecleaning and information retrieval. Also included here is a catalog titling apparatus. Usually sold as separate programs, such titlers (including this one) allow you to write "dummy" text files into the catalog to provide for fancy copyright notices and the like. This one allows every conceivable titling option including centered, left justified, inverse, flashing, bordered, blank lines, etc. The same program also runs the disk editing portion of Ampermanager, providing the occasional user with all the same power of the original, but in a heavily prompted environment.

Is it good value? Purchased separately, a complete Amper utility with the capability to manage other co-resident utilities automatically, a DOS editor, a track/sector editor, a catalog sorter, a catalog titler, and an Exec file writer might cost you nearly $200. Ampermanager sells for $22.50.

How Can You Use It?

With all this programming thrown in, what can you write to make use of the power of Ampermanager? Let me offer some of my own regular uses as suggestions. I like my Applesoft programs to format output on the screen. For me, a program menu exists as an array of strings MENU$(I) (I=1..MAX) waiting to be printed. This listing illustrates how I get a printout of each item starting at, say, line six with a number (right justified in case there are more than 10) and the menu item:

```
110 &WWDTH = PEEK($33):
    POS = (WWDTH/4) - 6
120 VTAB2: HOME: & PRINT USR
    WWDTH, "MENU"C
130 REM Prints in center of
    window
140 FOR I = 1 to MAX
150 &PRINT AT(POS,6 + I) USR
    3,I") "MENU$(I) :NEXT I
```

Besides the option of specifying a field of blanks as the "mask" in which to print, any string can be set up for that purpose. For instance, suppose we had money values which we wished to print in a column to two significant figures, with the decimals all lined up, and one space after each.

```
210 A$ = "**$0.00 <"
220 &?USR A$, 15.847!, 120!, 130.5
```

would produce

```
*$15.85 <
$120.00 <
$130.50 <
```

Note that extra characters are left in the string and that a $ sign is bumped to the left.

AT could be used to start the first entry anywhere on the 40 or 80 column screen (even though HTAB will not work on 80 columns). Unlike most such utilities, Ampermanager also handles exponents, social security numbers, and will justify left, right, or center in a given field.

I also use the &INPUT routine in preference to that of Applesoft. Messages can be sent, the ? at the end suppressed, and any line can be input regardless of whether it contains commas and colons.

Another facility that gets a lot of use by me is the &LET. With this one, we can substitute into the middle of a string, do unusual PEEKs, and set a variable equal to the position of one string in another. Closely related are &MOVE and &REV to move a string to any address or to reverse some string expression. Some valid statements include these:

```
410 &LET MID$(A$2,5) = "ABC"
420 &REV LEFT$(B$,4)
430 &WW = PEEK ($A2)
440 &A = IN#$(Offset, A$, B$)
450 &M$N = $AA75,4
460 REM This one points M$ to
    the string characters starting
    at the specified address, for
    the given length.
```

A lazy machine language programmer like me is always moving between the monitor and BASIC. &M does this in one swell foop. I also have provided statements which allow &POKE $300, $4C,$69,255 or &CALL $300 (uses the monitor G), and &PEEK($250). The latter is a function which can be used in &PRINT or &LET statements. On top of that you can convert between hex and decimal with &%decnum and &$hexnum, or indeed force printing of an expression in one of these formats by using the symbol in an &? statement. Speaking of the latter, it enhances the Applesoft PRINT statement in a number of ways, including printing multiple copies of a section of output, or outputting strings as they are stored, rather than in the usual format.

These are only a few of the Applesoft commands. There are a number for DOS as well including &CD (catalog deletes), &RELETE, &READ, &WRITE, &FMTDSK ON trk1 TO trk2, &TYPE and many others. I must admit that some of the more obscure commands were included only for the DOS or memory utilities in the driver program. However, you can read any disk sector, edit it and write it back to the disk, or dump and edit any page of RAM in the lower 48K of the memory.

Ampermanager for ProDOS

Even though Apple provides a (bug ridden) Convert utility to change DOS files to the ProDOS format, this is far from sufficient to produce a ProDOS version of my Ampermanager utilities. Ampermanager works intimately with DOS, making numerous calls directly into the machine language to open and close files and to manipulate the directory track. Since ProDOS is not just an upgrade but a completely new product, all of this had to be changed. Indeed, my resolution is to make no calls except through the MLI (Machine Language Interface), which is the only portion of ProDOS guaranteed not to change.

Consequently, the current release of Ampermanager (ProDOS version 2.0) lacks some of the utilities present in the DOS 3.3 version 1.5—which, however, does reside on the same disk. Gone (some forever, some temporarily) are Relete, Catalog Deletes, Free Space, Init, Put (Dos, VTOC or Cat) and some of the optional parameters associated with the 1.5 version DOS calls.

Still there are all the Applesoft extensions, with the AT function now enhanced to work with the Ultraterm—even if you have 48 rows on the screen. (The only guaranteed compatible eighty-column displays are BASIS, Apple *//e* and Ultraterm—others may work too.)

The Read/Dump/Zap/Write utilities are all rewritten to handle blocks as well as tracks and sectors (though for 40-column compatibility only half a block is displayed at a time), so you can still edit disks to your heart's content. Just type &READ6 to read block 6 and display it, or &READ,@$1104 to read and display track $11 sector 4 (the block number will show as well). There is no longer the distinction between reading on the directory track and elsewhere; and if you want to dump the next or last memory page, either an uppercase or lowercase + or - key will work on both the II+ and the *//e*. The same is true for reading the next or last blocks with the > and < keys; these changes eliminate the need to use the shift keys for this purpose.

Apart from the absence of some DOS commands and a slightly altered syntax for &READ, version 2.0 operates exactly as does version 1.5. I have not, however, rewritten the Applesoft driver utilities which go along with the 1.5 package, except for the EXEC file writer, as it used standard DOS calls. For now it's best to regard the ProDOS version as experi-

ANNOUNCING

COMPUTEREYES™

VIDEO IMAGES ON YOUR APPLE!

Finally — an inexpensive way to capture real-world images on your Apple's Hi-Res display! COMPUTEREYES™ is an innovative slow-scan device that connects between any standard video source (video tape recorder, video camera, videodisk, etc.) and the Apple's game I/O socket. Under simple software control, a b/w image is acquired in less than five seconds. A unique multi-scan mode also provides realistic grey-scale images. Hundreds of applications!

Package includes interface module, cable, complete easy-to-use software support on disk, owner's manual, and one year warranty. For 48K Apple II series and compatibles, with Applesoft and DOS 3.3. COMPUTEREYES™ is available from your dealer or direct from DIGITAL VISION for just $129.95 plus $4.00 S&H (USA).

Also available as a complete package including:
- COMPUTEREYES™
- Quality b/w video camera
- Connecting cable

for only $349.95 plus $9.00 S&H.

Demo disk available for only $10.00 postpaid.

Mass. residents add 5% sales tax. Mastercard, Visa accepted. To order, or for more information, write or call:

Screen dumps of actual COMPUTEREYES™ images.

DIGITAL VISION

DIGITAL VISION, INC.
14 Oak Street — Suite 2
Needham, MA 02192
(617) 444-9040

Circle 49 on Reader Service Card

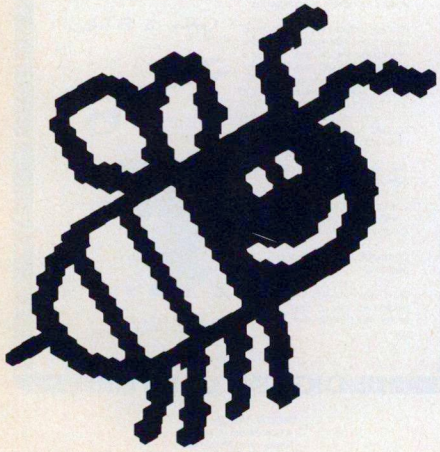
mental, and consider adding to the package when we have some feedback from users.

DOS 3.3 / ProDOS Hybrid Disks

Now let's consider the nature of the unusual disk upon which I have arranged all this programming. When you boot the Ampermanager 2.0 disk, you get ProDOS (so you need 64K unless you boot some other disk first.) If you select Ampermanager rather than just ProDOS at the first stage of the boot, you will be given a second set of choices after the working program is loaded. One of these is to switch to DOS 3.3—which resides on the same disk! Let me tell you how this is done.

In order to prepare a disk to run two operating systems, we need to insure that they will not conflict. In this case, the normal directory location for DOS 3.3 is track $11 while the corresponding ProDOS information is stored on track 0. However, both systems use track 0 for at least a portion of the code which does the initial startup of the system and finds and loads the appropriate DOS. Because we must have that ProDOS directory down on track 0, it will not do to have DOS 3.3 taking up its normal first three tracks. This forces us to design our system to "boot" ProDOS first and then somehow switch to DOS on demand. The solution to this is to transfer a copy of DOS 3.3 to ProDOS and start it up somewhat like a binary program.

A second difficulty lies in the fact that if we partition the disk between two operating systems, each must be informed that the other's territory is occupied so that it will not trespass. Fortunately, each system keeps a map of which areas are available, and it is a simple matter to change these maps to declare some territory out of bounds.

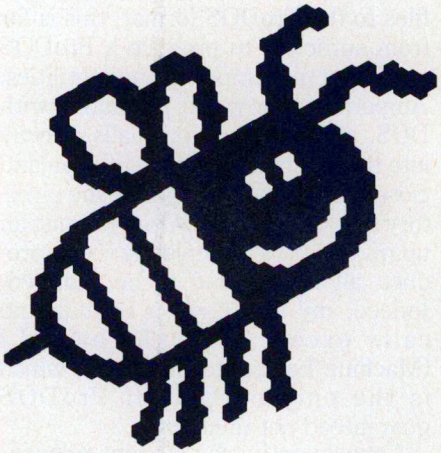


One thing which is not a problem is the format of the disks themselves. Even though DOS 3.3 uses sectors (256 bytes) and both ProDOS and Pascal use blocks (512 bytes), the actual information is all stored in sectors. Pascal and ProDOS simply read these two at a time. Consequently, the whole surface of the disk can be formatted by one of these three and still be read (at the lower level) by the other two. Here are the steps.

The Making of a Hybrid

First, I booted my /PRODOS developer's disk and then typed -FILER. (- is a "smart run" which decides whether to RUN, BRUN or EXEC when it sees what type of file you have named). The FILER utility has all the formatting, filing and copying commands you will ever need in ProDOS. Once there, I selected V)olume commands and chose the Format option. Part way through, it told me my drive 2 was running too fast, so I had to switch to drive 1 and restart the process.

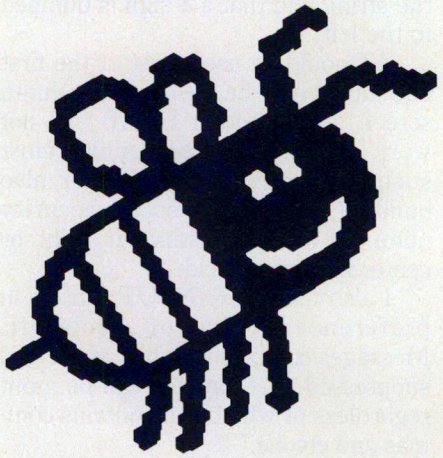
Next I pressed ESC a few times to go back to the main menu, Q)uit back to BASIC.SYSTEM, and typed BRUN AMPERMANAGER. I had already re-assembled and partially rewritten this utility to operate under ProDOS using Apple's fabulous new ProDOS EDASM. (A vast improvement on its ancestor, the venerable Toolkit assembler—for instance, it now has macros.)



Once AMPERMANAGER was installed, I reinserted the newly formatted AMPER and typed &READ6. In a flash I was looking at block 6 (the track bit map) on my screen. In this map, each track has one byte, and each of the eight bits of that byte indicate the status of one block on that track. I had already decided to reserve the last 18 tracks for DOS 3.3, so I typed Z11 and then put in a series of zeros to change all the FF's (all eight blocks on track free) to zeros (all eight used) from this point on . When done I pressed RET to inspect the rewritten block and seeing it seemed O.K. pressed !RET to write it back to the disk.

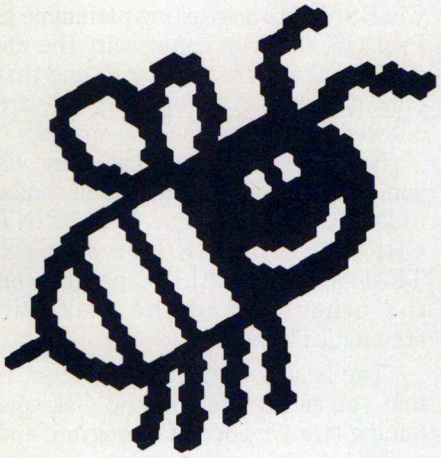
Checking this with &C (or CAT) showed the still-empty directory, but now the free space reported was a mere 129 blocks instead of 273 (seven are used for boot code and directory). The report of free space is one enhancement not needed on Ampermanager as it is included by ProDOS in both CAT and CATALOG (extended). Next, I transfered BASIC.SYSTEM, PRODOS, STARTUP, and AMPERMANAGER onto this new disk. All went into the lower numbered tracks as the higher ones were reserved.

Next I rebooted a DOS 3.3 master and brought up Ampermanager 1.5. I typed &PUTC to create a fresh 3.3 catalog on track $11 and then typed &READ0. (In the 1.5 version, this reads not block 0, but the first sector of track $11 called the VTOC.) Again I was staring at a track bit map—this was a typical 3.3 one which showed only tracks $11 and 0 through 2 used with everything else free (again 1 = free sector and 0 = used, but it takes two bytes/track as sixteen bits are needed. There are two unused zero bytes between each of these pairs.) I didn't want DOS to mess with the ProDOS portion so I typed Z43 and zeroed off bytes 43 to 78 to show tracks 0 to $10 as occupied. When done I again typed !RET and wrote this sector back to the disk.



At this point, a DOS 3.3 &C command or an &FRE will reveal 272 sectors free instead of the normal 496. I decided I wanted a few more, so I typed RUN THE DISK DIRECTOR and chose the item TRIM CATALOG which allowed me to release several of the track $11 sectors from directory use to general purpose use.

I then used FID to transfer all my working files from my old Ampermanager 1.5 master to the upper tracks of the new hybrid disk. I did not transfer the titles in the directory this way, because if I had, each would eat one sector of my precious space. When

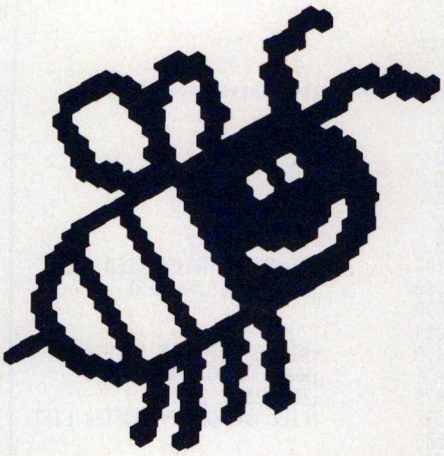


this was done, I re-entered THE DISK DIRECTOR and titled the catalog, putting in some new things to let buyers know it had ProDOS too, and updating my copyright notice. When I had a whole page of fancy titles ready to fill the screen (before a first keypress) whenever someone CATALOGed the disk, I quit the titling subroutine and wrote the appropriate sectors of Catalog track back (all automatic and with lots of warnings).

The last step here was to add a menu item to THE RECEPTIONIST which would re-boot the whole system whenever anyone made the choice "switch" to ProDOS. I had also added a BASIC program with the standard "HELLO" name containing one line to EXEC the real startup program. This portion of the disk was now complete, and a few sectors remained for possible program alterations.

DOS to DOS Switching

The next task was to get the ProDOS to DOS 3.3 changeover to work without changing disks. To achieve this, I first entered the monitor via &M and then moved a copy of DOS to $2100 with the command 2100< 9D00 .BFFFM. I saved a copy of this on a DOS 3.3 disk via BSAVE DOS 3.3, A$2100, L$2300. I was now finished with this part, so I re-booted ProDOS. My first task was to get DOS



3.3 as a ProDOS file, so I typed -CONVERT and in response to my answers to the prompts, this program obligingly made me a copy of this file and stored it on a ProDOS disk.

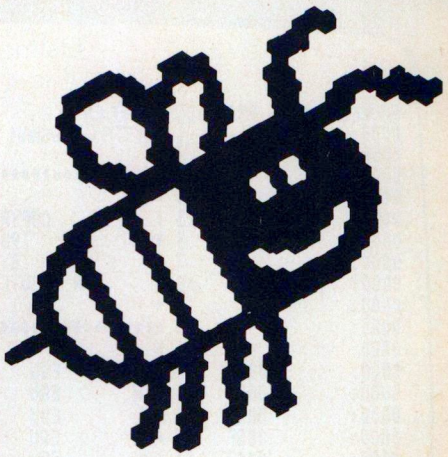
I now had just about everything ready, but I wanted a "live" DOS 3.3 which would cold-start itself, set up page 3 and the I/O hooks and run the HELLO program. Furthermore, I knew that I could not simply BLOAD DOS 3.3 into its final location at $9D00 for I would in that process overwrite the code doing the loading. Besides, ProDOS must be turned off before starting a new system program.

The trick here is to turn DOS 3.3 into a .SYSTEM program. Whenever you "-" a program of this type, ProDOS does a BLOAD at $2000, turns itself off and does a JMP to $2000 in the expectation of finding there a routine which can relocate its working part up to high memory and jump into the coldstart for that system.

Thinking ahead, I had left $2000 to $20FF for just such a routine, so I entered EDASM and wrote the routine in Listing 1 below. Note that all EQUates to the DOS portion of memory refer to the relocated 3.3 code.

The portion called MOVER sets up the addresses used by the monitor MOVE routine, and once this is called (in line 45) the code residing at $2100 through $42FF will be back in place from $9D00 to $BFFF.

Next comes DOSPREP which makes it 'live' by telling it that INIT was



Can you make an idea fly?

We're looking for Apple II programmers who can help us get our graphic ideas off the ground. If you create smooth animation, concise, well-documented code and work fluently in assembly language and basic, we'd like to talk to you. Please contact our Vice President of Product Development at this address:

Springboard Software, Inc.
7807 Creekridge Circle
Minneapolis, MN 55435
(612) 944-3912.

PROGRAMMERS WANTED.

Circle 54 on Reader Service Card

Listing 1

```
SOURCE   FILE #01 =)DOS3.3MOVER
0000:              1 *SYSTEM PROGRAM TO SWITCH INTO DOS 3.3 FROM PRODOS
0000:              2 *
0000:              3 ************************************************
0000:              4 *
0000:              5 *          COPYRIGHT
0000:              6 *            1984
0000:              7 *             BY
0000:              8 *       R.  SUTCLIFFE
0000:              9 *
0000:             10 ************************************************
0000:             11 *
0000:       003C  12 A1L      EQU   $3C          ;SOURCE
0000:       003D  13 A1H      EQU   $3D
0000:       003E  14 A2L      EQU   $3E          ;END OF SOURCE
0000:       003F  15 A2H      EQU   $3F
0000:       0042  16 A4L      EQU   $42          ;DESTINATION
0000:       0043  17 A4H      EQU   $43
0000:       2100  18 DOSCOPY  EQU   $2100        ;START OF TRANSFERRED CODE
0000:       9D00  19 DOSSTRT  EQU   $9D00        ;DOS3.3 BEGINS HERE
0000:       9D0D  20 DOSP1    EQU   $9D0D
0000:       9D84  21 DOSCOLD  EQU   $9D84
0000:       A095  22 FNBLK    EQU   $A095        ;BLANKS FILENAME BUFFER
0000:       AA5F  23 DOSCMD   EQU   $AA5F        ;WHAT TO DO WHEN BOOTED
0000:       AA75  24 PFNBUF   EQU   $AA75        ;FILE NAME BUFFER
0000:       B5BC  25 FMPLST   EQU   $B5BC        ;FILE MANAGER PARAMETER LIST
0000:       B5BB  26 FMPCMD   EQU   $B5BB
0000:       B7C2  27 SETDSP   EQU   $B7C2
0000:       FE2C  28 MOVE     EQU   $FE2C        ;MONITOR MOVE ROUTINE
0000:       FE89  29 SETKBD   EQU   $FE89
0000:       FE93  30 SETVID   EQU   $FE93
0000:             31 *
----- NEXT OBJECT FILE NAME IS DOS3.3MOVER.0
2000:       2000  32          ORG   $2000        ;FOR ALL SYSTEM PROGRAMS
2000:             33 *
2000:A0 9D        34 MOVER    LDY   #<DOSSTRT    ;SET PARAMS FOR MONITOR MOVE
2002:84 43        35          STY   A4H          ;IN PAGE ZERO
2004:A0 21        36          LDY   #<DOSCOPY
2006:84 3D        37          STY   A1H
2008:A0 43        38          LDY   #$43
200A:84 3F        39          STY   A2H
200C:A0 FF        40          LDY   #$FF
200E:84 3E        41          STY   A2L
2010:C8           42          INY                ;NOW 0
2011:84 3C        43          STY   A1L
2013:84 42        44          STY   A4L
2015:20 2C FE     45          JSR   MOVE         ;AND MOVE DOS 3.3 IN
2018:20 89 FE     46          JSR   SETKBD
201B:20 93 FE     47          JSR   SETVID
201E:             48 *
201E:A9 0B        49 DOSPREP  LDA   #$0B         ;INIT COMMAND MAKES IT LIVE
2020:8D BB B5     50          STA   FMPCMD
2023:A9 00        51          LDA   #$00         ;RUN COMMAND
2025:8D 5F AA     52          STA   DOSCMD       ;FOR GREETING PROGRAM
2028:20 95 A0     53 FILENAME JSR   FNBLK
202B:A2 00        54          LDX   #0
202D:BD 3F 20     55 FNAME1   LDA   NAME,X       ;PUT IN OUR NAME
2030:48           56          PHA
2031:09 80        57          ORA   #$80         ;SET HI BIT ALWAYS
2033:9D 75 AA     58          STA   PFNBUF,X
2036:68           59          PLA                ;DONE IF AT HI BIT SET
2037:30 03  203C  60          BMI   FIN          ;YES
2039:E8           61          INX                ;NOT YET
203A:D0 F1  202D  62          BNE   FNAME1       ;ALWAYS
203C:4C 84 9D     63 FIN      JMP   DOSCOLD      ;GO START IT UP
203F:             64 *
203F:             65 *FILENAME TO RUN GOES HERE
203F:48 45 4C 4C  66 NAME     DCI   /HELLO/
2044:             67 *
```

its last command (i.e. you've just been booted, baby) and that the next one to do is RUN (though this could be BRUN or EXEC with the appropriate number).

FNBLK blanks out the primary file name buffer and the rest of FILENAME puts in 'HELLO' ready to be found there when the DOS becomes active. At this point when the JMP to DOSCOLD is done, the DOS will believe it is freshly booted from a disk and does a normal startup.

Once this portion was assembled and checked, I BLOADed it at $2000 and then BLOADed DOS 3.3 at $2100. Rather than just BSAVE this as a Binary file at this point, I decided to implement it as a .System file so that prior to execution ProDOS would be properly turned off. So I typed first CREATE DOS 3.3.SYSTEM, TSYS and then BSAVE DOS 3.3.SYSTEM, TSYS, A$2000, E$43FF. Note that you BSAVE to an existing pathname in ProDOS, that you can specify the file type by T and the end by E, and that only alphanumerics and the "." are allowed in a pathname—no spaces.

Now it's done—we can exit via some Applesoft program menu choice to DOS 3.3 with the line 1000 PRINT CHR$(4)"-/AMPER/DOS 3.3.SYSTEM" and the "HELLO" program on the other half of the disk will automatically be run.

The beauty of this arrangement is that you can create a hybrid disk containing two versions of a program and write instructions to wait for user input before deciding on which environment to use. Then, when that environment is chosen, it is in every way a normal one—neither DOS nor ProDOS is patched in any way to achieve this effect. By the way, since the DOS 3.3 always does RUN HELLO,S6,D1, you can "-" DOS 3.3.SYSTEM from drive 2 to start up something else (with a program called HELLO) from Drive 1. However, AMPER must be in Drive 1 when switching between the versions on the hybrid disk itself. Of course, the assembly language program could be modified to BRUN or EXEC or to RUN some other program than HELLO, or no program at all.

Ampermanager 1.5 Updates

Document:

On page 42 change pad S5 to pad X6 and pin 27 to pin 24.

Copies shipped before January 1984: Change line 1050 of "The Executive" to start with &INPUT rather than just INPUT.

A.P.P.L.E. Consultants

John A. Hall, Jr	Apple //e(East)	(617) 241-9050
Cecil Fretwell	ProDOS/Apple //e(Midwest)	(319) 236-0961
Bob Huelsdonk	Apple ///(West)	(206) 362-4910
Harry T. Hanson	Apple ///(East)	(201) 467-0712
Carmen Picardi	Apple ///(East)	(603) 895-4447
Stuart Yusem	Apple ///(East)	(609) 822-9306
Tim Smith	Apple ///(Midwest)	(309) 734-7451
Neil Clemmons	Agriculture	(314) 968-9618
Erv Edge	Assembly(Alaska)	(907) 337-7620
Dennis Baumgarte	Assembly(East)	(914) 298-8756
Murray Arnow	CP/M(Midwest)	(312) 867-8000
Kevin Talbot	CP/M(West)	(206) 232-7058
Steven E. Bair	Data Bases	(913) 782-5901
Jerry Rogan	Data Bases, Business	(319) 332-4166
William P. Steinberg	DOS(East)	(212) 921-0663
Art Schumer	DOS(North)	(701) 282-7907
Lee Meador	DOS(South)	(817) 469-6019
William Reynolds III	DOS(West)	(503) 472-4028
Ted Perry	Education	(916) 485-1690
Cornelis Bongers	DOS,Applesoft(Europe)	31-010-525511 Ext.3275
Alan Robinson	FORTRAN	(503) 757-8234
Jim Eatherly	Games	(202) 232-6046
Tedd Sperling	Geoscience	(517) 393-1296
Stan Axelrod	Hardware Design, I/O Control	(415) 563-1840
Tom Deal, M.D.	Handicapped Applications	(206) 634-2847
Randy Halvorson	Hardware Interfacing	(206) 783-9040
Ed So	Hi-Res	(408) 267-3448
Stowe Keller	Mathematics	(804) 971-6876
Fred Merchant, M.D.	Medical Group	(415) 921-7792
Stephen F. Lee	Music	(704) 289-4102
Tom Woteki	Pascal(East)	(202) 547-0984
Tim Smith	Pascal(Midwest)	(309) 734-7451
Paul E. Sleeper	Pascal(West)	(415) 494-1915
Bruce E. Sargent	Pascal(West)	(206) 488-1161
Dennis R. Cohen	Pascal(West)	(213) 956-8559
Earl Keyser	Pilot Language	(612) 429-5570
Randy Warrens	Pilot Language	(608) 837-9566
Jeff Sandys	Logo(West)	(206) 784-3317
Hersch Pilloff	Stock Market	(301) 292-3100
Jim Hockenhull	Video Interfacing	(509) 334-4137
George L. Hefter	VisiCalc(East)	(617) 256-3082
Vince Aderente	VisiCalc(North)	(312) 289-9685
Dr. Bob Richard	VisiCalc(South)	(504) 466-0894
Brian Abrahams	VisiCalc(South Africa)	(Durban) 315442
Michal Ginsberg	Word Processing	(805) 735-8210

Call -A.P.P.L.E. consultants are unpaid volunteers and perform a valuable service. Before placing a call, please make sure you know the time zone you are calling to. No calls after 9pm, please.

We are always looking for more consultants. If you see an area that needs coverage, please drop us a note in the mail with your qualifications and availability.

Apple PugetSound Program Library Exchange

Officers and Directors

Dick Hubert	President
Bob Huelsdonk	Vice President
Dave Lingwood	Secretary
Bob Clardy	Treasurer
Dave DeGroot	Director
Val J. Golding	Director
Don Williams	Director

Administration

Dick Hubert	Executive Director
Don Steele	Asst to Executive Director
Dave O'Leary	Comptroller
Steve Johnson	Technical Staff
Charles Stillman	Manager of Education Services

Departments

Order Desk
Tracing
Member Information
Education, Classes
Retail store
Dealer services
Marketing
Technical support
Testing and Evaluation
Operations

Member Services

24 hour toll-free order line:
1-800-426-3667
(except Washington state)
Orders ONLY
Information not available

Programming/Technical Assistance
Call-A.P.P.L.E. HOTLINE
(206) 872-9004
9am to 3pm - 6pm to 10pm
(Pacific time)
7 days per week

A.P.P.L.E. Crate Bulletin Board
(206) 872-6789

Meetings
7:30pm third Tuesday of each month
A.P.P.L.E. Office
21246 68th Ave. S., Kent
(except as noted)

Membership Information
is available on postpaid tear-off card inside

A.P.P.L.E. on the SOURCE
A special section of The Source is open only to A.P.P.L.E. members who may download Call-A.P.P.L.E. programs. To access, type: CALL-A.P.P.L.E. from Source command mode. To register, send mail to ST1570 with your name, membership number, expiration date and Source account number.
Printed in U.S.A.

THE Spreadsheet Sheet—Update

In the June issue of *Call -A.P.P.L.E.*, "THE Spreadsheet Sheet—Data Analysis #7" included some incorrect data in the printed tables. In Table 7.1, cell H69 should have read +K49. Tables 7.2 and 7.4 are correct, but Table 7.3 requires extensive modification. If you wish to have a copy of the corrected version, please send a written request to our editorial office.

Due to space limitations, *THE Spreadsheet Sheet* does not appear in this issue.

IF YOU'RE READY TO GO
& YOUR Apple IIe SAYS NO
YOU NEED
TYPE AHEAD BUFFER
TAB-1

TAB-1 GIVES YOU CONTROL OF YOUR APPLE IIe. The simple addition of TRAC SYSTEM's TAB-1 buffer to your Apple IIe increases the efficiency of your computer dramatically. TAB-1 eliminates the chance that you might lose characters while typing ahead. With TAB-1 installed, you no longer have to sit and wait for your Apple IIe to catch up!

TOP QUALITY, HIGH TECH CONSTRUCTION. Your TAB-1 circuit board is manufactured to meet rigorous quality standards to insure lasting, trouble-free compatibility with your Apple IIe.

SIMPLE INSTALLATION. TAB-1 is easily installed in your Apple IIe. Clear, step-by-step, instructions make installation a simple procedure.

TRAC SYSTEM'S TAB-1 is the lowest priced buffer available for your Apple IIe. Order your TAB-1 today!

*Registered Trademark of Apple Computer, Inc. TAB-1 is manufactured and exclusively distributed by TRAC SYSTEMS CORPORATION

ONLY $45

Order Blank

Please send me _____ TAB-1 Type Ahead Buffers at $45 each _____
California Residents Please Add Sales Tax _____
Shipping and Handling charge for Total Order $3.00
Total Enclosed for Order _____
NAME: _____
ADDRESS: _____
CITY: _____ STATE: _____ ZIP: _____
Send order and check or money order to : TRAC SYSTEMS CORP., 444 North 3rd Street, Suite 201, Sacramento, CA 94814. Please allow 4 weeks for delivery.

Circle 60 on Reader Service Card

July 1984

MEMBER BILLBOARD

Don Elman

New A.P.P.L.E. Catalog

BY THE time you read this, all A.P.P.L.E. members should have received the new, improved, and greatly expanded 1984 A.P.P.L.E. Products Catalog. If yours has not yet arrived by July 15th, write or call our operations department.

Among the enhanced features are improved readability, more photos, better organization by category, and over 40 brand new entries. In particular, you'll want to look through the section devoted to *//c*–compatible products, and note that we now have a page dedicated to items for the Macintosh and Lisa. Also, if you know any non–members who might be interested in A.P.P.L.E.'s services, be sure to let them know about the new member specials described in the catalog.

Catalog Updates and Backorders

With this issue of *Call –A.P.P.L.E.* we also begin a regular practice of announcing approximately 10 new and updated products each month. Now you won't have to wait as long to find out which items are constantly being added to our stock. When the next semi–annual catalog is issued, all of these products, and more, will be included.

For most products we keep an ample supply on hand so that each order can be filled as soon as it is received. Many of the items, however, are so new or so popular that we have trouble keeping them in stock, and occasionally there is a delay in filling your orders. Since we know how frustrating it is to wait, we are constantly striving to reduce or eliminate backorders. In some cases this situation just can't be avoided, and we sincerely thank all affected members for their patience.

The New A.P.P.L.E. Crate Bulletin Board System

For those of you with an interest in telecommunication, we'd like to correct a typo from last month and then give you some very good news. In the June *Call –A.P.P.L.E.*, we accidentally listed an incorrect phone number for our A.P.P.L.E. Crate Bulletin Board. The correct number is **(206) 872–6789.**

If you haven't yet had the pleasure of using a computerized bulletin board system (BBS), you should know that this growing phenomenon provides a unique medium of communication between people—unlike either letters or telephones. For many users it can be one of the most interesting applications of your computer. All you need, in addition to a basic computer system, is a piece of hardware called a "modem" that connects to your telephone line, and the proper software for sending and receiving remote signals. The bulletin board itself consists of another computer and modem on the other end of the line, with more sophisticated software and the capability of storing and responding to incoming messages.

Now for the news. The former system software used for the A.P.P.L.E. Crate BBS, which was capable only of storing and retrieving a single list of messages, has been replaced by a more advanced, customized system. As modified by our own Darrin Massena, the GBBS][system from G. S. Software has a "room–oriented" approach. Messages can be left in each of several "rooms," and each room has its own distinctive title or orientation, such as "Macintosh/Lisa," "Apple *//c*," or "ProDOS." If you wish to scan existing messages, you can choose the room or rooms that fit your interests. There is also an E–mail option for sending private messages to a particular person.

The system storage is on a Corvus 20–megabyte hard disk, so there should be plenty of room for expansion. A unique feature of this system is "new mail scan," which gives you only those messages which have been posted since the last time you were on the system. Program downloading will also be available. Users of the old system should be advised that all previous messages and passwords are unavailable, so you should expect to set up a new password when you next sign on. The SYSOP (system operator) is looking forward to your comments and suggestions.

New A.P.P.L.E. Products

For Membership & Orders only,
24 hr. toll free 1-800-426-3667

Apple][+ , IIe, IIc

VOLKSFILER

by Glen Hausler

SVOLKS
$12.50
N/A
2 oz.

VOLKSFILER is a freeform electronic card file program where each "card" is represented by your Apple screen. You may enter information wherever you want it on the screen. When you recall the information later, it will appear exactly as you entered it. The first word on each screen is used as a sort key. The fast search command automatically scans the entire contents of every screen for the word you want. Automatic data compression assures efficient use of disk storage. VOLKSFILER is easily mastered and is best suited for informal note-taking and other jobs where a conventional database structure would be too restrictive.

HIGHER GRAPHICS II

by Synergistic Software

SHIGRA
$15.00
$35.00
8 oz.

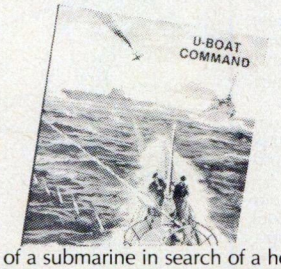
At last we offer you a full-function shape table editor, so you may use the Applesoft DRAW command, the great "sleeper" of HIRES graphics. With this program, you may edit from scratch, view and change shapes from any existing table, or even pull together shapes from several different tables. The Applesoft DRAW command and shape tables offer big advantages in programs where space is at a premium. The disk comes with a wide variety of sample shapes, among them a complete character set. The latter is especially handy for labelling HIRES graphs horizontally, vertically, or even upside down and backwards! You will find that DRAW is even fast enough to do simple animation.

THE FONT ALBUM

by Andrew Wicker

SFTALB
$12.50
N/A
2 oz.

Here is a brand new set of over forty fonts for use with Higher Text. Find just the one for the program you are writing. You'll find a wide variety of thin, thick, slanted, shadowed, and stylized typefaces. There is even one patterned after that voracious little arcade game character who has consumed so many quarters! Some fonts include special shapes, creatures, tanks, and other objects that can be used for programming character graphics in creating games.



U-BOAT COMMAND

by Synergistic Software

SUBCOM
$12.50
$29.95
8 oz.

You are the commander of a submarine in search of a heavily defended enemy aircraft carrier and it's making a break for neutral waters. Watch the sonar scope, air supply, fuel... DEPTH CHARGES! . . . steady . . . steady . . . FIRE ONE! Arcade style strategic war game.

AMPERMANAGER Version 2.0

by Rick Sutcliffe

SAMP
$22.50
N/A
8 oz.

This new version of our popular Ampersand utility package now works under ProDOS as well as DOS 3.3. Read or write disk blocks at will in ProDOS, and any Track-Sector pair in DOS 3.3. Useful utilities include improved input supporting lower case and allowing commas and colons, quick variable swapping, formatted printing (print using), and many more. See "The Northern Spy" in this issue for further details.

STARGAZER'S GUIDE

by Synergistic Software

SSTAGU
$15.00
$30.00
8 oz.

This program gives you an entertaining way to learn the night skies and recognize the major constellations. Included is a handy reference booklet, a mini-astronomy course complete with explanations of nomenclature, tables of stellar statistics, and folklore about the constellations.

BOLO

by Synergistic Software

Product Code	SBOLO
A.P.P.L.E. Price	$15.00
Sug. Retail	$34.95
Ship. Wt.	8 oz.

On a battle¢torn planet, only you survived an alien ambush. Alone and vastly outnumbered, your task is to seek out and destroy the rapidly growing alien tank force. Multi-leveled Arcade Style Tank Conflict.

Hardware

PANASONIC KX-1091 PRINTER

HPANPT
$369.00
$495.00
20 lbs.

This new printer features a 10 inch wide carriage, 120 character per second speed, Pica, Elite and condensed fonts, and high density bit graphics (1920 dots/line). The outstanding print is near letter quality, with proportional spacing, subscripting and superscripting as needed. The friction/tractor feed can handle up to 3-part paper. It is Epson RX-80, and IBM-PC dot matrix/graphics compatible. The print head has a 150 million character life and the ribbon a 4 million character life. The 8 bit parallel printer and 1K buffer are standard. A super printer at a reasonable price.

VIDEX UNIPRINT PARALLEL PRINTER INTERFACE

HVIDIN
$74.00
$89.00
1 lb. 6 oz.

A single graphics printer interface compatible with over 25 printers! Works with Apple DOS, ProDOS, Pascal, Logo, and CP/M in Text mode. Special commands are available to control page width, length, and left margin. In graphics mode HIRES graphs and pictures (Double HIRES on the *IIe*) are easily printed (even in color on the IDS Prism printer). Parallel cable and extensive documentation included.

Apple III

III E-Z Pieces by Haba Systems

SEZPIE
$250.00
$295.00
2 lbs.

Word Processor, spreadsheet and database manager for the Apple *III*. Data can easily be moved among the three applications. Learning is simplified since all applications use similar commands. This program was developed by Rupert Lissner, author of Quickfile, and is the Apple *III* version of Apple Works produced for the Apple][. The program runs on any Apple *III* and is Catalyst compatible. Manual and 3 diskettes.

CALENDAR PAK *III* by TAASCOM (The All American Software Company)

SCALPK
$79.00
$95.00
1 lb.

This remarkable personal calendar and scratchpad program runs in background and can be called up inside virtually any Apple *III* program! You can actually view and change appointments without leaving the program you are running. You could be using your word processor program, look at your calendar, add an appointment or write yourself a note, all without ever leaving the word processor. When an appointment time is reached, the program you are running is interrupted with a message. The SCRATCHPAD feature even allows you to move information between Calendar Pak and your other applications. This may be the most practical personal calendar program available for any microcomputer. Runs on any Apple *III* and is Catalyst compatible.

Mac

TITLE MASTER

STITMA
$20.00
N/A
8 oz.

When you want to use the 48 point and 72 point fonts for large titles, do you suffer from the JAGGIES? If so, you need TITLE MASTER. This collection of smoothed and refined characters for six of the MacIntosh fonts makes clean looking large titles for signs, notices and documents. These characters are not available directly from the keyboard. Titles can be constructed by using cut and paste from the Title Master files. A document is included which explains the efficient use of this technique. Fonts supported are: Monaco, Venice, Chicago, London, Athens, and San Francisco.

Catalyst is a trademark of Quark Engineering. Quickfile and Apple Works are trademarks of Apple Computer Inc.

Apple® Programming Utilities

☐ GPLE™ (Supports DOS 3.3 *and* ProDOS)

GLOBAL PROGRAM LINE EDITOR by NEIL KONZEN
$49.95: Includes Peeks/Pokes Chart & Tip Book #7

THE CLASSIC APPLE PROGRAM LINE EDITOR: GPLE lets you edit 40- or 80-column Applesoft program lines *FAST* without awkward cursor-tracing or "Escape editing". Compatible with Double-Take's 2 way scrolling.

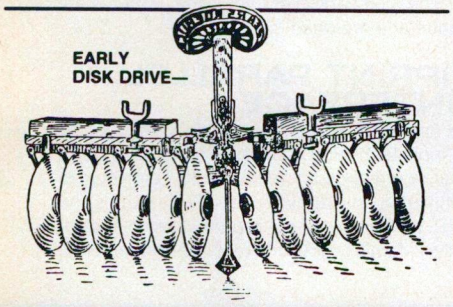
INSERT & DELETE: GPLE works like an in-memory word processor for Applesoft program lines. Simply jump the cursor to the change-point and insert or delete text. No need to trace to the end of a line before hitting Return.

GLOBAL SEARCH & REPLACE: Find any word or variable in your programs, *FAST*. For example, find all lines containing a GOSUB, or all occurrences of variable XY. REPLACE ANY VARIABLE or word with any other. For example, change all X's to ABC's, or all "Horses" to "Cows".

DEFINABLE ESC FUNCTIONS: Define ESC plus any key to perform any task. For example, **ESC-1** can catalog drive 1, or **ESC-N** could type an entire phrase or subroutine. *Anything* you want, *whenever* you want.

GPLE DOS MOVER: Move DOS 3.3 & GPLE above main memory for an EXTRA **10K** of programming space.

PLUS APPLE TIP BOOK #7: Learn more about your Apple! Includes all-new useful GPLE tips and tricks.



☐ BEAGLE BASIC™ (REQ. 64K)

APPLESOFT ENHANCER by MARK SIMONSEN
$34.95: Includes Peeks/Pokes Chart & Tip Book #6

RENAME ANY APPLESOFT COMMAND or Error Message to anything you want. For program clarification, encryption/protection or even foreign translation. Example:

10 POUR X=1 TO 3: ECRIVEZ "BONJOUR": ENSUITE

RAM Applesoft is *better* Applesoft! Beagle Basic replaces those obsolete cassette commands (SHLOAD, etc.), with powerful new commands that you can USE—

ELSE follows Applesoft If-Then statements, like this:

IF X=2 THEN PRINT "YES": *ELSE* PRINT "NO"

HSCRN reads the color of a hi-res dot for collision testing, **SWAP** exchanges variable values, **TONE** writes music without messy Pokes or Calls, **SCRL** scrolls text in *either* direction, **TXT2** lets Text Page 2 act exactly like Page 1...

GOTO AND GOSUB may precede variables, as in "GOSUB FIX" or "GOTO 4+X". Escape-mode indicated by a special ESCAPE CURSOR. Replace those awkward Graphics screen-switch pokes with one-word commands. Change your ctrl-G Beep to any tone you want. **INVERSE REM STATEMENTS** too! GPLE/Double-Take compatible.

☐ DOS BOSS™

DOS 3.3 EDITOR by BERT KERSEY & JACK CASSIDY
$24.00: Includes Peeks/Pokes Chart & Tip Book #2

RENAME DOS COMMANDS and Error Messages—DOS 3.3's "Catalog" can be "Cat"; DOS's cryptic "Syntax Error" can be "Oops" or almost *anything* you want it to be.

PROTECT YOUR PROGRAMS. Unauthorized Save-attempts can produce "Not Copyable" message, or *any* message. **List-Prevention** and other useful Apple tips and tricks. Plus one-key program-execution from catalog.

CUSTOMIZE DOS. Change Disk Volume headings to your message or title. Omit or alter catalog file codes. Fascinating documentation, tips & educational experiments.

ANYONE USING YOUR DISKS (booted or not) will be forced to use DOS formatted the way YOU designed it.

☐ DOUBLE-TAKE ← DOS 3.3 / ProDOS
☐ DOUBLE-TAKE/PRO™

2-WAY SCROLL/MULTI-UTILITY by MARK SIMONSEN
$34.95: Includes Peeks/Pokes Chart & Tips Chart #1

2-WAY SCROLLING: Listings & Catalogs scroll Up *and* Down, making file names and program lines easier to access. Change Catalog or List scroll-direction with Apple's Arrow keys. Machine Language and Hex/Ascii dumps scroll two-ways too. All features support 80-column display.

BETTER LIST FORMAT: Each Applesoft program statement lists on a new line for *FAST* program tracing & de-bugging. Printer-compatible in any column-width.

```
A$="DOGFOOD"
X=3.14159
Y=255
```
← **VARIABLE-DISPLAY:** prints all of a program's strings and variables with their current values.

```
A$: 100 200 250
X: 10 20 3000
Y: 10 40 55 60
```
← **CROSS-REFERENCE:** Sorts & displays line numbers where each variable & string appears.

AUTO-LINE-NUMBER, instant Hex/Dec Converter, better Renumber/Append, Program Stats, Eliminate/Redefine Cursor, Free Space-On-Disk... All GPLE/Pronto compatible.

2-WAY VIDEO SCROLLING

```
]LIST
10 HGR2
   : FOR Y=0 TO 191
   : POKE 228, C
   : C=C+1/9-256*(C=255)
20 REM "Double-Take
   optionally lists each
   Applesoft program
   statement on a new
   line, making pro-
   gram code much
   easier to read."
30 HPLOT 0,Y TO 279,Y
   : NEXT Y
   : POKE 2053,58
   : GOTO 10
```

High-Speed DOS

☐ PRONTO-DOS™

HIGH-SPEED DOS/DOS-MOVER by TOM WEISHAAR
$29.50: Includes Peeks & Pokes Chart

TRIPLES THE SPEED of disk access and frees 10,000 bytes of extra programmable memory by moving DOS 3.3.

Function	Normal	Pronto
BLOAD HI-RES IMAGE	10 sec.	3 sec.
LOAD 60-SECTOR PROGRAM	16 sec.	4 sec.
SAVE 60-SECTOR PROGRAM	24 sec.	9 sec.

(Text-files no change) **Bload language cards** at *triple speed*. Create bootable high-speed disks with the normal INIT command. Compatible with *all* commands, GPLE, Double-Take, DOS Boss, DiskQuik & most unprotected programs.

MOVE DOS 3.3 above main memory to free *10,000 bytes* of memory for your programs (64K required to move DOS).

15 EXTRA SECTORS per disk. Catalog Free-Space is displayed on the screen every time you Catalog a disk.

NEW TYPE-COMMAND ("TYPE filename") prints the contents of any Text File on-screen or to your printer.

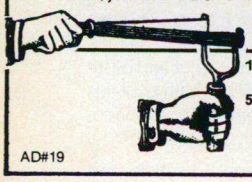
☐ DISKQUIK™ (Requires IIe with 128K)

DISK EMULATOR by HARRY BRUCE & GENE HITE
$29.50: Includes Peeks & Pokes Chart
Requires IIe with EXTENDED 80-column card (128K)

ACTS LIKE A DISK DRIVE in Slot 3, but super-fast and silent! Enjoy many of the benefits of another drive at 1/10th the cost. Catalog with "CATALOG, S3" command. Access all kind of files in RAM with normal DOS commands.

SILENT AND FAST: Since no moving parts are involved, DiskQuik operates at super-high speeds. See to believe! Your Apple IIe's Extended 80-column Card (required) holds about half the amount of data as a 5¼" floppy!

MANY USES: For example, load often-used files like FID into RAM when you boot up, so they are always available when you need them. Copy files from RAM onto disk and vice versa, just as if a disk drive were connected to slot #3.



COMPATIBLE with all normal DOS procedures.

```
1234 TEXT: HOME: NORMAL:
     PRINT CHR$(21)
5678 R=INT(RND(1)*10): N(R)=
     N(R)+1: VTAB R+9: HTAB
     40: PRINT CHR$ (124); SPC
     (N(R)); CHR$(R+65); :
     IF PEEK(36) THEN 5678
```

AD#19

Multi-Utilities

☐ UTILITY CITY™

21 PROGRAMMING UTILITIES by BERT KERSEY
$29.50: Includes Peeks/Pokes Chart & Tip Book #3

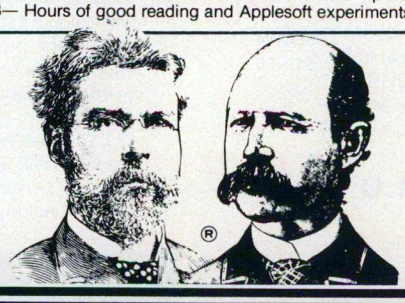
LIST FORMATTER prints each Applesoft program statement on a new line. For-Next Loops are indented with printer Page Breaks. A great Applesoft program de-bugger.

MULTI-COLUMN CATALOGS to your printer, with or without sector and file codes. Organize your disk library.

INVISIBLE AND TRICK catalog File Names. Put invisible functioning commands in Applesoft programs too.

21 UTILITIES TOTAL, including auto-post Run-number & Date in programs, alphabetize/store info on disk, convert dec to hex or Int to FP, protect and append programs, dump 40-column text to printer. And More.

LEARN PROGRAMMING TRICKS: LIST-able programs and informative documentation. Includes Tip Book #3— Hours of good reading and Applesoft experiments.

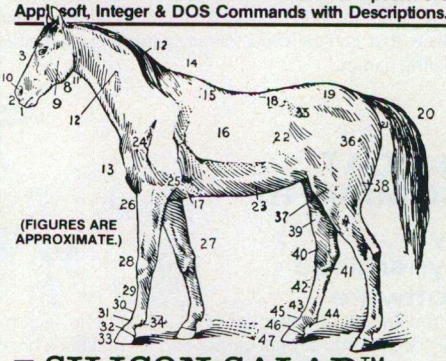


Beagle Bros
Micro Software Inc.

3990 OLD TOWN AVENUE, SUITE 102C
SAN DIEGO, CA 92110 / 619-296-6400

Attention Applers: Most Apple dealers and software stores have Beagle Bros products on their shelves. If you can't find the disk you want, bug the manager—he can have *any* of our products in his store for you within a couple of days.

FREE APPLE COMMAND CHART: Each SILICON SALAD and TIP DISK #1 comes with an 11x17 poster of all Applesoft, Integer & DOS Commands with Descriptions.



☐ SILICON SALAD™

WITH TIP DISK #2 by BERT KERSEY & MARK SIMONSEN
$24.95: With Peeks/Pokes Chart & Apple Command Chart

MANY MINI-UTILITIES: Program Splitter makes room for hi-res pix in large Applesoft programs, **Disk Scanner** finds bad disk sectors, **Key-Clicker** adds subtle sound as you type, **DOS-Killer** adds two tracks of space to your disks, **2-Track Cat** allows up to 210 DOS 3.3 file names per disk, **Text Imprinter** converts text-screen text into hi-res text, **Onerr Tell Me** prints the appropriate error message but continues program execution, **Text Screen Formatter** formats and converts text layouts into Print statements... plus much more Apple wizardry from the boys at Beagle Bros.

MORE TIPS ON DISK: Over 100 programs from Beagle Bros Tip Books 5, 6 and 7; *and* from Tip Chart #1.

TWO-LINERS TOO: From our customers around the world—and elsewhere. New tricks for your old Apple!

☐ TIP DISK #1™

100 TIP BOOK PROGRAMS ON DISK by BERT KERSEY
$20.00: With Peeks/Pokes Chart & Apple Command Chart

100 LISTABLE PROGRAMS from Beagle Tip Books 1-4. Make your Apple do things it's never done! All 100 programs are LISTable and changeable for experimentation. Two-Liners too, plus a free *Apple Command Chart*.

COMING JULY 84 Apple® Graphics Utilities

☐ BEAGLE GRAPHICS™

DOUBLE HI-RES GRAPHICS by MARK SIMONSEN
$59.95: Includes Peeks & Pokes Chart
Requires Apple IIc (or IIe with EXTENDED 80-col. card).

560-WIDE WITH 16 COLORS! Beagle Graphics gives your 128K-Apple (required) a full range of colors plus *double* high resolution—560 x 192 pixels. All Applesoft hi-res functions, including shape tables, are supported.

NEW COMMANDS let you draw fast circles, ellipses and rectangles from the keyboard or from your programs.

FAST COLOR FILL fills any outline with one of 16 solid colors or 256 color mixes (usable in your programs).

PROGRAM & PICTURE-CONVERTERS change your existing Applesoft programs *and pictures* to double hi-res. "Double-ize" *Apple Mechanic* shape table programs too!

DOUBLE-PLOT PROGRAM lets you draw and manipulate pictures in double hi-res. Add type to pix too. Move sections of images to either page. Save pictures to disk.

HI-RES TRICKS: Amazing stuff— any portion of a picture may be rotated, flopped, moved, inverted, superimposed, scrunched or even SAVED to disk. Saving image-portions conserves disk space.

```
10 PRINT CHR$
(ASC (CHR$ (ASC
(CHR$ (ASC ("F") /
(ASC ("P") / 8))))):
GOTO 10
```

COMING JULY 84

☐ TRIPLE-DUMP™

SINGLE/DOUBLE HI-RES "PRINT-ANYTHING" UTILITY
$39.95: Includes Peeks & Pokes Chart

PRINT ANY SCREEN IMAGE on your dot-matrix (graphics-capable) printer— Hi-Res, Lo-Res, **Double Hi-Res,** Medium-Res, as well as 40 **and 80-Column Text**. All print functions may be used in your Applesoft programs and disks. Don't settle for a "locked up" printer-dump program.

SPECIAL EFFECTS: Crop, rotate, enlarge, distort, invert... Preview pictures on the screen before you print.

BANNER MAKER: Make impressive 8"-high signs and banners for your family, home or office. Type any message—no length limit—and let your printer do the work!

☐ ALPHA PLOT™

STANDARD HI-RES GRAPHICS UTILITY
by BERT KERSEY & JACK CASSIDY
$39.50: Includes Peeks/Pokes Chart & Tip Book #4
(Alpha Plot offers 6-color 280-pixel resolution and requires only 48K; see Beagle Graphics for 128K double hi-res.)

DRAW IN HI-RES on both pages using easy keyboard commands. Pre-view lines before plotting. Use solid or mixed colors and Reverse (background opposite). One-key-stroke circles, boxes and ellipses, filled or not. All pictures are Save-able to disk for access by your Applesoft programs.

COMPRESS HI-RES DATA to 1/3 disk-space, allowing 3-times the number of hi-res pictures per disk (avg. figures).

MANIPULATE IMAGES: Superimpose pictures or *relocate* sections of images anywhere on either hi-res page.

HI-RES TYPE: Add variable-size color & b/w text to your pictures. Type *anywhere* with no htab/vtab limits. Type sideways too, for Charts & Graphs. Includes Tip Book #4.

☐ FLEX TYPE™

FLEXIBLE-TEXT UTILITY by MARK SIMONSEN
$29.50: Includes Peeks & Pokes Chart

PUT VARIABLE-WIDTH TEXT on both hi-res screens with normal Applesoft commands (including HTAB 1-70). Normal, expanded & compressed text with no extra hardware. (70-column text requires b/w monitor, not a tv).

COMBINE TEXT & GRAPHICS. Run existing Applesoft programs with Flex. GPLE/Double-Take compatible.

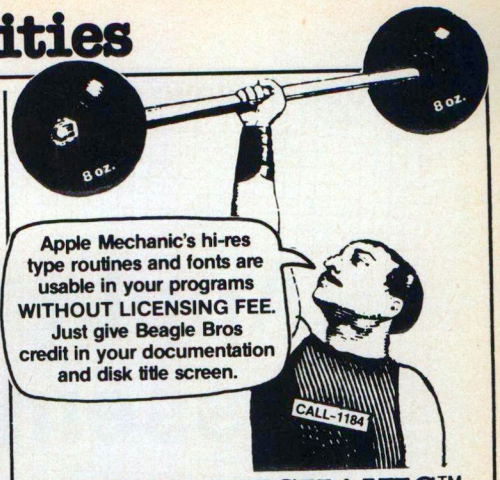
DOS TOOL KIT® font compatibility. Access up to 9 fonts in memory. Text editor lets you redesign text characters.

☐ FRAME-UP™

FAST APPLE DISPLAY UTILITY by TOM WEISHAAR
$29.50: Includes Peeks & Pokes Chart

PROFESSIONAL PRESENTATIONS: Turn existing Hi-Res, Lo-Res & Text pix into Apple "slide shows". *FAST* 2½-sec. hi-res loading! Paddle or Keyboard-advance frames.

UNATTENDED SHOWS optional— each picture arranged and pre-programmed to display 1 to 99 seconds. Text Screen Editor creates black & white text "slides" ; lets you key-in type "live" during shows. Mail copies of presentations-on-disk to friends & associates (or home to Mom!).


8 oz.
8 oz.
Apple Mechanic's hi-res type routines and fonts are usable in your programs WITHOUT LICENSING FEE. Just give Beagle Bros credit in your documentation and disk title screen.
CALL-1184

☐ APPLE MECHANIC™

HI-RES SHAPE EDITOR & FONTS by BERT KERSEY
$29.50: Includes Peeks/Pokes Chart & Tip Book #5

SHAPE EDITOR: Keyboard-draw hi-res shapes for animation in your Applesoft programs. Easy "List & Learn" Applesoft demos teach how to do hi-res animation as well as professional-looking hi-res Charts and Graphs.

HI-RES FONTS: Access & create proportionally-spaced hi-res type; each character totally re-definable. Six complete fonts are included on the disk (4 large & 2 small).

MORE: Useful music, text and hi-res tricks for your Applesoft programs. Clear educational documentation and step-by-step instructions for writing graphics programs.

APPLE MECHANIC HI-RES ☐ TYPEFACES™

26 FONTS FOR APPLE MECHANIC by BERT KERSEY
$20.00: Includes Peeks & Pokes Chart

26 NEW FONTS for use with Apple Mechanic programs. Many different sizes and typestyles, both ordinary and *Artistic*. Every character—from A to Z to "★" to "□"—of every typeface—from "Ace" to "Zooloo"—is re-definable to suit your needs. All type is *proportionally spaced* for a more professional appearance. People *do* notice the difference!

Disk Librarian

☐ FATCAT™ NEW!

PERSONAL DISK LiBRARY by ALAN BIRD
$34.95: Includes Peeks & Pokes Chart

MULTIPLE-DISK CATALOG: FATCAT reads all of your DOS 3.3 and ProDOS disks into one or more "Master Catalogs" that can be searched, sorted and printed. Get organized, and find out what's *really* on all of those floppies!

ALPHABETIZE FILE NAMES: FATCAT lets you sort your DOS 3.3 and ProDOS disk catalogs alphabetically, by file name, type, etc., to make files easier to find. Once arranged, a new disk directory may be written to the disk, so your files are in the order you want when you CATALOG. This feature alone is worth the price of FATCAT!

```
10 REM BEAGLE CARD FILE
20 HGR: HGR2: POKE 232, 120: POKE 233, 64: POKE
16504, 7: SCALE=80: P=16: X=99: FOR R=0 TO 31:
P=P*-1: POKE 230, 48+P
30 FOR Y=0 TO 1: ROT=ABS(64*Y-R): HCOLOR=3:
FOR A=1 TO 25: DRAW 1 AT X+2*A, X: NEXT
40 HCOLOR=0: DRAW 1 AT X, X: ROT=32: DRAW 1:
DRAW 1 AT X, X: NEXT Y, R
50 FOR A=0 TO 20: FOR B=0 TO 1: POKE 49237-B,0:
X=PEEK(49200): FOR C=1 TO 6*A
60 NEXT C, B, A: GOTO 50
```

Beagle Bag!

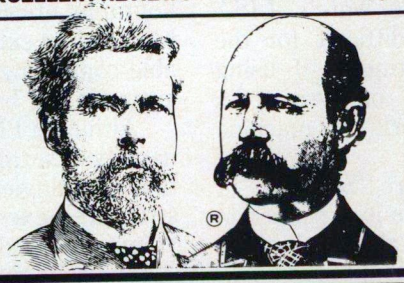
☐ BEAGLE BAG™

12 APPLE GAMES ON DISK by BERT KERSEY
$29.50: Includes Peeks & Pokes Chart
Apple software customers recently voted BEAGLE BAG to Softalk's "Most Popular Software" list for 1983.

COMPARE BEAGLE BAG with any one-game locked-up disk on the market today. All 12 games are a blast, the price is right, the instructions are crystal clear, *and* the disk is COPYABLE. You can even *change* the programs or list them to learn programming tricks by seeing how they work.

TWELVE GAMES that *last*— TextTrain, Wowzo, Magic Pack, Buzzword, Slippery Digits, and many more...

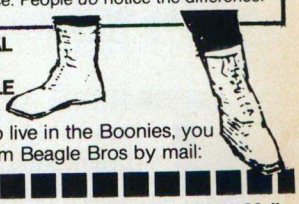
EXCELLENT REVIEWS—Read Jan-83 *Softalk*, pg.148.



Beagle Bros
Micro Software Inc.

3990 OLD TOWN AVENUE, SUITE 102C
SAN DIEGO, CA 92110 / 619-296-6400

Attention Dealers: Beagle Bros products rank consistently high on software best-seller charts, and captured FOUR spots on *Softalk's* 1983 Top-30 Most-Popular List (all categories). Telephone any Apple software distributor for fast delivery.



GOTO YOUR LOCAL APPLE SOFTWARE STORE FOR BEAGLE BROS PRODUCTS.

Or, if you happen to live in the Boonies, you can buy directly from Beagle Bros by mail:

RUSH the following disks by First Class Mail:

☐ Alpha Plot	$39.50	☐ Fatcat	$34.95
☐ Apple Mechanic	29.50	☐ Flex Type	29.50
☐ A.M.Typefaces	20.00	☐ Frame-Up	29.50
☐ Beagle Bag	29.50	☐ GPLE	49.95
☐ Beagle Basic	34.95	☐ ProntoDOS	29.50
☐ Beagle Graphics†	59.95	☐ Silicon Salad	24.95
☐ DiskQuik	29.50	☐ Tip Disk #1	20.00
☐ DOS Boss	24.00	☐ Triple Dump†	39.95
☐ Double-Take/3.3	34.95	☐ Utility City	29.50
☐ Double-Take/Pro	34.95	☐	

†Available July 1, 1984

AT YOUR APPLE DEALER NOW!
Or order directly from Beagle Bros.

Visa/MasterCard or COD, call TOLL-FREE
Orders only / ALL 50 STATES / 24 Hours a Day
1-800-227-3800 ext. 1607

OR mail U.S.Check, Money-Order or Visa/MCard# to:
BEAGLE BROS, Dept. M
3990 Old Town Avenue, Suite 102C, San Diego, Ca 92110

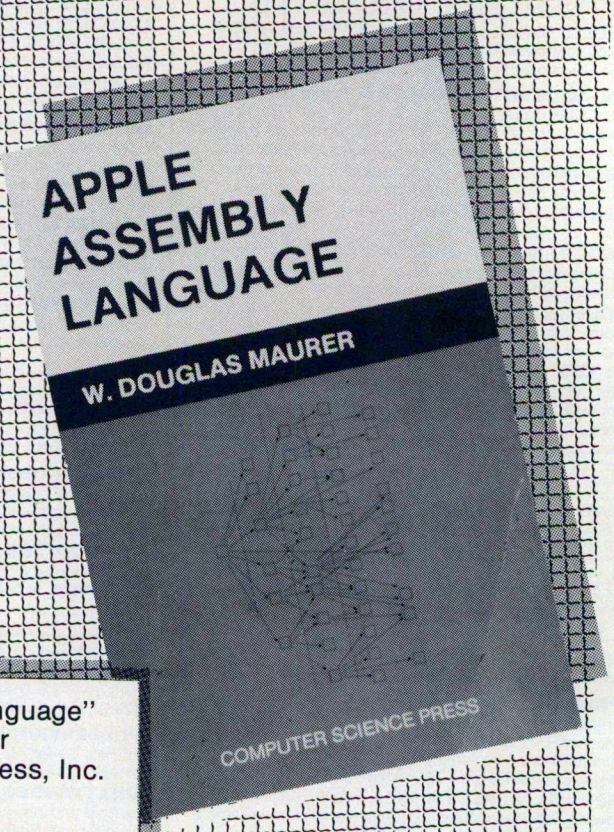
NAME _____
ADDRESS _____
_____
STATE & ZIP _____
VISA/MC# _____
EXPIRATION ___/___ PLEASE SIGN IN MARGIN→

Add $1.50 First Class Shipping, Any-Size Order.
Overseas add $4.00. COD add $3.00. California add 6%.
ALL ORDERS SHIPPED IMMEDIATELY.

"APPLE" is a Registered Trade Mark of You-Know-Who.

Book Review:

Apple Assembly Language



"Apple Assembly Language"
by W. Douglas Maurer
Computer Science Press, Inc.
11 Taft Court
Rockville, MD 20850
6x8" paperback,402 pages.
$19.95

Lee Webster

THERE ARE a number of available books on assembly language—some are even Apple oriented. This one, written by a faculty member at George Washington University, is one of the few I have seen that is designed as a *textbook* for secondary school or college level, although it would also be good for individual study.

Apple Assembly Language assumes you have already had at least one course in computers, as it makes numerous references to how things work in BASIC. It is also designed to be used with LazerWare's LISA assembler, although it shouldn't be difficult for the reader to make the necessary adjustments if using another assembler. The emphasis is on writing in assembly language with limited work in machine language (such as in debugging).

The book is organized into 100 short "Sections" of 3–4 pages each. Essentially one principle is explained in each section followed by 3 exercises. For about half of the exercises, the answers are given in the back of the book.

The early section topics include discussions of number systems and registers, loading and storing (in memory cells), addressing (including subscripted variables), addition/subtraction, branching, loops, offsets, I/O, multiplication by 2 and 10, and subroutines for multiplication/division. Discussion of complete programs doesn't start until Section 42, and the first problem that requires computer use comes in section 50. Otherwise, most of the work involves hand assembly and desk checking. Later sections cover logical AND/OR, bit testing, stacks, hardware stacks (one of few discussions of hardware items), BCD numbers, interrupts, queues and polling, real-time programming, indexed addressing, strings, two-dimensional arrays, simulators, and interpreters. Sections are included (when needed) on the use of the LISA assembler.

The Appendix contains lists of BASIC terms, hex-dec conversion tables, 6502 instructions, LISA pseudo-ops, Apple character codes, Apple monitor subroutines, and monitor commands. There is also a bibliography of assembly language books for other computers, and an extensive 14 page index. A diskette is available containing all the programs and fragments of programs in the book.

Although I have not read many other assembly language books, this is the best I have seen. The short sections on single topics are well written, and the well-structured exercises provide a chance to see how well you understand each point.

For self-study use, it would be helpful to have the answers to *all* of the exercises rather than just some of them. I also question whether it is good to put off the first "hands-on" computer work until well into the book. It might be better to do a little memory manipulation (maybe from the monitor level) earlier, just to get the feel of what is happening and to begin understanding hex notation. I must confess, I've tried to learn assembly language from other books with limited success. This one looks like it will do the trick, and I'd recommend that you take a look at it, too.

A.P.P.L.E. Products

(Available to members only)

SOFTWARE

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	SAMP	AMPERMANAGER	$22.50	0	4
☐	SAPF	APFILE	$18.50	0	2
☐	SLFR	APGRAPHICS "The Filler"	$17.00	0	4
☐	SAM3	APMAIL Version 3.0	$25.00	0	3
☐	SAPMPR	APMAIL.PRO	$35.00	0	3
☐	SAPMUP	APMAIL.PRO UPGRADE	$10.00	0	1
☐	SCAR	APPLESOFT CARPENTER II	$18.50	0	4
☐	SCARUP	CARPENTER II UPGRADE	$10.00	0	1
☐	SAPT	APTEST	$18.50	0	2
☐	SMACC	BIG MAC.C	$40.00	1	0
☐	SMLC	BIG MAC.LC	$22.50	0	7
☐	SMAU	BIG MAC UPDATE	$10.00	0	3
☐	SDIC	DICE	$12.50	0	2
☐	STITWR	DIRECTORY TITLE WRITER	$15.00	0	2
☐	SD6B	DISKPAK 6B-16	$18.50	0	1
☐	SEAP	E-Z PILOT	$28.50	0	4
☐	SCPM	FLASHCARD/SSD CP/M DISK	$30.00	0	1
☐	SDOS	FLASHCARD/SSD DOS DISK	$30.00	0	1
☐	SPAS	FLASHCARD/SSD PASCAL DISK	$30.00	0	1
☐	SHFI	HIGHER FONTS #1	$13.00	0	1
☐	SHTP	HIGHER TEXT PLUS	$27.50	0	4
☐	SIMAGI	IMAGINATOR	$60.00	2	0
☐	SDISRE	LIFESAVER	$21.00	0	5
☐	SQDRAW	LISA QUICK DRAW DEMO	$39.50	0	2
☐	SCALST	MICRO PRO: CALCSTAR	$115.00	2	0
☐	SDATST	MICRO PRO: DATASTAR	$200.00	5	0
☐	SINFST	MICRO PRO: INFOSTAR	$295.00	8	0
☐	SMAILM	MICRO PRO: MAILMERGE	$175.00	3	0
☐	SREPST	MICRO PRO: REPORTSTAR	$205.00	6	0
☐	SSPEST	MICRO PRO: SPELLSTAR	$147.00	2	0
☐	SSTIND	MICRO PRO: STARINDEX	$115.00	2	0
☐	SSUPSO	MICRO PRO: SUPERSORT	$147.00	3	0
☐	SMUL	MULTI-TRIEVE	$50.00	0	9
☐	SMULUP	MULTI-TRIEVE 1.3 Upgrade	$10.00	0	5
☐	SPGE	PROGRAM GLOBAL EDITOR	$18.50	0	2
☐	SPLE	PROGRAM LINE EDITOR	$15.00	0	3
☐	SPZAP	PROZAP	$25.00	0	3
☐	SPDS	PUBLIC DOMAIN SOFTWARE	$5.00	0	1
☐	SS70	SOFT SEVENTY	$20.00	0	3
☐	SSYM	SYMBOL SYMON	$20.00	0	3
☐	STSS	THE SPREADSHEET	$65.00	2	0
☐	STSC	THE SPREADSHEET 1.0 to 2.0 FILE CONVERTER	$5.00	0	1
☐	SCHEER	/// CHEERS (1 year)	$40.00	0	4
☐	SCHE1	/// CHEERS (Summer 84)	$12.50	0	1
☐	SCHE2	/// CHEERS (Fall 84)	$12.50	0	1
☐	SCHE3	/// CHEERS (Winter 84)	$12.50	0	1
☐	SCHE4	/// CHEERS (Spring 85)	$12.50	0	1
☐	STUR	TURTLESOFT	$22.50	0	8
☐	SDAN	VERBATIM DISK DRIVE ANALYZER	$32.00	0	9
☐	SPWSPR	WORDSTAR PROFESSIONAL	$450.00	9	0
☐	SZARII	ZARDAX II (Apple II family)	$160.00	2	0

VALUESOFT

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	SCATKI	CATER KILLER	$12.50	0	2
☐	SDOSAN	DISK ANALYZER 1.3	$12.50	0	2
☐	SFARWO	FARWORD	$12.50	0	2
☐	SGRAFM	GRAFMAGIC	$12.50	0	2
☐	SGRABU	GRAPHICS BUILDER	$12.50	0	2
☐	SMICWT	MICROWRITER *IIe*	$12.50	0	2
☐	SMORTG	MORTGAGES, DEPREC. AND FINANCIAL CALCULATIONS	$12.50	0	2
☐	SGRADE	SCHOOLHOUSE GRADER	$12.50	0	2
☐	SSCRAM	SCRAMBLER	$12.50	0	2
☐	SWICWA	WICKERWARE #1	$12.50	0	2
☐	SZAPTI	ZAPT	$12.50	0	2

GAMES

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	SCCO	COSMIC COMBAT	$12.50	0	1
☐	SD7	DISKPAK 7	$17.50	0	2
☐	SEAMON	EAMON DISK	$5.00	0	1
☐	SKOALA	KOALAGRAMS SPELLING I	$31.50	0	3
☐	SMIC	MICROBE	$30.00	1	0
☐	SMCU	MUMMY'S CURSE	$12.50	0	1
☐	SSPI	SPIDER EATER	$24.00	0	3
☐	STHB	THUNDERBOMBS	$16.50	0	4
☐	STRA	TRANSYLVANIA	$25.50	0	4

A.P.P.L.E. LIBRARY

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	MM-A80	*Call-A.P.P.L.E.* March/April 80	$2.00	0	8
☐	MMAY80	*Call-A.P.P.L.E.* May 80	$2.00	0	8
☐	MJUN80	*Call-A.P.P.L.E.* June 80	$2.00	0	8
☐	MN-D80	*Call-A.P.P.L.E.* Nov/Dec 80	$2.00	0	8
☐	MJAN81	*Call-A.P.P.L.E.* January 81	$2.50	0	8
☐	MFEB81	*Call-A.P.P.L.E.* February 81	$2.50	0	8
☐	MM-A81	*Call-A.P.P.L.E.* March/April 81	$2.50	0	8
☐	MMAY81	*Call-A.P.P.L.E.* May 81	$2.50	0	8
☐	MJUN81	*Call-A.P.P.L.E.* June 81	$2.50	0	8
☐	MJ-A81	*Call-A.P.P.L.E.* July/Aug 81	$2.50	0	8
☐	MSEP81	*Call-A.P.P.L.E.* September 81	$2.50	0	8
☐	MOCT81	*Call-A.P.P.L.E.* October 81	$2.50	0	8
☐	MN-D81	*Call-A.P.P.L.E.* Nov/Dec 81	$2.50	0	8
☐	MJAN82	*Call-A.P.P.L.E.* January 82	$2.50	0	8
☐	MFEB82	*Call-A.P.P.L.E.* February 82	$2.50	0	8
☐	MMAR82	*Call-A.P.P.L.E.* March 82	$2.50	0	8
☐	MAPR82	*Call-A.P.P.L.E.* April 82	$2.50	0	8
☐	MMAY82	*Call-A.P.P.L.E.* May 82	$2.50	0	8
☐	MJUN82	*Call-A.P.P.L.E.* June 82	$2.50	0	8
☐	MJUL82	*Call-A.P.P.L.E.* July 82	$2.50	0	8
☐	MAUG82	*Call-A.P.P.L.E.* August 82	$2.50	0	8
☐	MSEP82	*Call-A.P.P.L.E.* September 82	$2.50	0	8
☐	MOCT82	*Call-A.P.P.L.E.* October 82	$2.50	0	8
☐	MNOV82	*Call-A.P.P.L.E.* November 82	$2.50	0	8
☐	MDEC82	*Call-A.P.P.L.E.* December 82	$2.50	0	8
☐	MJAN83	*Call-A.P.P.L.E.* January 83	$2.50	0	8
☐	MFEB83	*Call-A.P.P.L.E.* February 83	$2.50	0	8
☐	MMAR83	*Call-A.P.P.L.E.* March 83	$2.50	0	8
☐	MAPR83	*Call-A.P.P.L.E.* April 83	$2.50	0	8
☐	MMAY83	*Call-A.P.P.L.E.* May 83	$2.50	0	8
☐	MJUN83	*Call-A.P.P.L.E.* June 83	$2.50	0	8
☐	MJUL83	*Call-A.P.P.L.E.* July 83	$2.50	0	8
☐	MAUG83	*Call-A.P.P.L.E.* August 83	$2.50	0	8
☐	MSEP83	*Call-A.P.P.L.E.* September 83	$2.50	0	8
☐	MOCT83	*Call-A.P.P.L.E.* October 83	$2.50	0	8
☐	MNOV83	*Call-A.P.P.L.E.* November 83	$2.50	0	8
☐	MDEC83	*Call-A.P.P.L.E.* December 83	$2.50	0	8
☐	MJAN84	*Call-A.P.P.L.E.* January 84	$3.00	0	8
☐	MFEB84	*Call-A.P.P.L.E.* February 84	$3.00	0	8
☐	MMAR84	*Call-A.P.P.L.E.* March 84	$3.00	0	8
☐	MAPR84	*Call-A.P.P.L.E.* April 84	$3.00	0	8
☐	MMAY84	*Call-A.P.P.L.E.* May 84	$3.00	0	8
☐	MJUN84	*Call-A.P.P.L.E.* June 84	$3.00	0	8
☐	MJUL84	*Call-A.P.P.L.E.* July 84	$3.00	0	8
☐	PACD	APPLE II CIRCUIT DESCRIPTION	$18.50	3	0
☐	PAO1	APPLE ORCHARD #1	$1.00	1	0
☐	PASL	ASSEMBLY LINES: THE BOOK	$17.00	1	0
☐	PBAD	BENEATH APPLE DOS	$17.00	1	0
☐	PAPP2C	GOING PLACES WITH THE APPLE *IIc*	$3.25	0	7
☐	PAPALM	THE APPLE ALMANAC	$17.50	2	0
☐	PAPCON	THE APPLE CONNECTION	$12.95	2	0
☐	PAPHDB	THE APPLE II BASIC HANDBOOK	$12.95	1	0
☐	PELA	THE ELEMENTARY APPLE	$11.00	1	0
☐	PAUTGU	SOFTWARE AUTHOR'S GUIDE	$17.50	2	0
☐	PDBII	UNDERSTANDING DBASE II	$19.95	2	0
☐	PID1	IN DEPTH VOL #1 (APPLESOFT)	$7.00	1	0
☐	SIN1	IN DEPTH DISK #1 (APPLESOFT)	$16.50	0	1
☐	PID2	IN DEPTH VOL #2 (PASCAL)	$11.00	2	0
☐	SIN2	IN DEPTH DISK #2 (PASCAL)	$20.00	0	3
☐	PID3	IN DEPTH VOL #3 (DOS)	$13.50	2	0
☐	SIN3	IN DEPTH DISK #3 (DOS)	$15.00	0	2
☐	PID4	IN DEPTH VOL #4 (APPLEWRITER *IIe*)	$12.50	1	0
☐	PIMAGS	IMAGINATOR SOURCE LISTING	$30.00	1	0
☐	PKID	KIDS AND THE APPLE	$14.00	2	0
☐	PMACIN	INTRODUCING THE APPLE MACINTOSH	$10.50	1	0
☐	PMAS	MICRO ON THE APPLE (3 VOL SET)	$40.00	3	0
☐	PMA1	MICRO ON THE APPLE(VOL 1)	$15.00	1	0
☐	PMA2	MICRO ON THE APPLE(VOL 2)	$15.00	1	0
☐	PMA3	MICRO ON THE APPLE(VOL 3)	$15.00	1	0
☐	PPK2	VOL II 1979 Compendium	$10.00	1	0
☐	PPK3	VOL III 1980 Compendium	$15.00	2	0
☐	PPK4	BOUND SET *Call-A.P.P.L.E.* 1981	$18.50	4	0
☐	PPK5	BOUND SET *Call-A.P.P.L.E.* 1982	$18.50	6	0
☐	PPK6	BOUND SET *Call-A.P.P.L.E.* 1983	$18.50	6	0

A.P.P.L.E. LIBRARY

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	PU65	USING 6502 ASSEMBLY LANGUAGE	$17.00	1	0
☐	PVIS	THE POWER OF VISICALC VOL I	$8.50	1	0
☐	PVI2	VOL II	$8.50	1	0
☐	PWII	WOZPAK II	$10.00	1	0
☐	PWTA	WOZ TALKS (VHS video tape)	$27.50	1	0
☐	PWWA	WHAT'S WHERE IN THE APPLE	$17.00	2	0
☐	PWW2	THE GUIDE	$8.00	2	0
☐	PPLGRO	COMPUTER PLAYGROUND	$8.50	1	0
☐	PCP1	BEST OF CIDER PRESS	$1.00	0	7

HARDWARE

	CODE	PRODUCT	PRICE	LBS.	OZ.
☐	HSRMON	AMBER MONITOR (12")	$100.00	17	0
☐	HAPC	AP CONTROLLER	$85.00	0	4
☐	HAKK	AP COOL KIT	$25.00	2	0
☐	HAPD	AP DRIVE	$225.00	6	0
☐	HLC1	AP.LCASE 1	$30.00	0	2
☐	HLC1A	AP.LCASE 1 w/ DICE	$37.50	0	3
☐	HLC2	AP.LCASE 2	$25.00	0	1
☐	HLC2B	AP.LCASE 2 w/ DICE	$30.00	0	2
☐	HAUD	APPLESURANCE	$85.00	1	0
☐	H80C	APPLE *IIe* 80 COLUMN CARD	$120.00	0	4
☐	HBASF	BASF 3.5" MICRO-FLOPPY (BOX 5)	$22.50	0	7
☐	HDBA	DISK BANKS	$4.50	1	0
☐	HDJA	DISKETTE JACKETS (100)	$10.00	1	8
☐	HGEN10	DISKETTES-GENERIC (BULK 10)	$19.00	0	8
		(set of 100)	$150.00	5	0
☐	HVEB	DISKETTES-VERBATIM (10 BOX)	$25.00	1	0
☐	HMPC	DUMPLING GX	$130.00	1	0
☐	HMPD	DUMPLING 16	$180.00	1	0
☐	HDUST	DUST COVER	$8.00	0	7
☐	HSSD	FLASHCARD/SSD 294K	$525.00	2	0
☐	HSSS	FLASHCARD/SSD 147K	$345.00	2	0
☐	HSSU	FLASHCARD 147K UPGRADE KIT	$200.00	0	2
☐	HFLS	FLIP SORT DISK ORGANIZER	$22.50	3	0
☐	HGRAAP	GRAPHITTI-APPLE DMP	$90.00	2	0
☐	HGRACI	GRAPHITTI-C.ITOH/NEC/PRO	$90.00	2	0
☐	HGRAEP	GRAPHITTI-EPSON/ PANASON.	$90.00	2	0
☐	HGRAID	GRAPHITTI-IDS	$90.00	2	0
☐	HGRAOK	GRAPHITTI-OKIDATA	$90.00	2	0
☐	HHOC	HOME CONTROLLER	$40.00	0	8
☐	HPAD	KOALA PAD TOUCH TABLET	$95.00	2	0
☐	HKOA2C	KOALA PAD TOUCH TABLET FOR *IIc*	$95.00	2	0
☐	HJMACH	MACH III JOYSTICK FOR][+	$44.00	1	0
☐	HJOMA3	MACH III JOYSTICK FOR *IIe*	$49.00	1	0
☐	HJOYST	MACH II JOYSTICK FOR][+	$34.00	1	0
☐	HMACPK	MACPAK	$64.95	6	0
☐	HMAGPH	MAGICPHONE	$180.00	0	10
☐	HDMEMO	MEMOREX 3.5" MICRO-FLOPPY (BOX 10)	$44.00	1	0
☐	HTDSDR	MICRO DRIVE][+, *IIe*	$210.00	5	0
☐	HTDS2C	MICRO DRIVE *IIc*	$225.00	5	0
☐	HMOBII	MOCKINGBOARD][+, *IIe*	$165.00	20	
☐	HMOB2C	MOCKINGBOARD *IIc*	$165.00	3	0
☐	HWSP1	MODEM II AND THE SOURCE	$255.00	0	0
☐	HMOD2E	MULTIMODEM *IIe*	$275.00	3	0
☐	HPADDL	PADDLE-ADAPPLE	$25.00	0	5
☐	HPANMO	PANASONIC COLOR MONITOR	$279.00	23	0
☐	HPANPR	PANASONIC KX1090 PRINTER	$325.00	20	0
☐	HPANRI	PANASONIC RIBBONS	$14.00	0	5
☐	HSERIN	PANASONIC RS232 SERIAL OPTION	$85.00	1	0
☐	HPOST2	POSTER (Feb84) Cover	$9.95	0	2
☐	HPOWSU	POWER SUPPLY (II,][+,*IIe*)	$85.00	3	0
☐	HPRC	PRINTER RIBBON PROW/NEC/AP/DMP	$4.95	0	3
☐	HTRC	PRINTER RIBBON TALLY MT 160L	$12.50	0	7
☐	HTRI	PRINTER RIBBON TALLY MT 180L	$17.00	1	0
☐	HPRMOD	PROMODEM 1200	$450.00	6	0
☐	HPRO2C	PROMODEM 1200 *IIc*	$450.00	7	0

Use Order Form on Reverse

A.P.P.L.E. Products (Available to members only)

HARDWARE

CODE	PRODUCT	PRICE	LBS.	OZ.
☐ HPRPAK	PRINTPAK	$34.00	3	0
☐ HPRT	PROWRITER PRINTER	$380.00	24	0
☐ HP11	PROTOTYPING BOARD 11"	$21.00	0	5
☐ HP16	PROMETHEUS 16K RAM CARD	$55.00	0	8
☐ HP28	PROMETHEUS 128K RAM CARD	$295.00	1	0
☐ HRIGNO	RIGHT NOTCH	$15.00	0	2
☐ HRPT	REPEATERRR	$25.00	0	5
☐ HSOFTC	SOFTCASE	$39.00	3	0
☐ HSUPMO	SUP'R MOD	$60.00	2	0
☐ HFAN	SUPER FAN II	$79.00	3	0
☐ HTSHL	T SHIRT:LARGE	$10.95	0	5
☐ HTSHM	T SHIRT:MEDIUM	$10.95	0	5
☐ HTSHS	T SHIRT:SMALL	$10.95	0	5
☐ HTSHX	T SHIRT:XLARGE	$10.95	0	5
☐ HTR120	TRANSTAR 120 PRINTER	$559.00	23	0
☐ HPAG	TYPING AID	$13.00	2	0
☐ HULTRA	THE ULTRATERM	$249.00	2	0
☐ HURB	ULTRA ROM BOARD/EDITOR	$149.00	2	0
☐ HVIE80	VIEWMASTER 80 COL. CARD	$149.00	2	0
☐ HVRC	VERSACARD (NO CABLE)	$150.00	1	0
☐ HVRC1	w/ 6' Parallel Cable	$160.00	2	0
☐ HVRC2F	w/ 6' Female Serial Cable	$160.00	2	0
☐ HVRC2M	w/ 6' Male Serial Cable	$160.00	2	0
☐ HVRC3F	w/ 1' Female Serial Cable	$160.00	2	0
☐ HVRC3M	w/ 1' Male Serial Cable	$160.00	2	0
	Cables Only			
☐ HVPC	6' Parallel Cable	$20.00	0	4
☐ HVSLF	6' Female Serial Cable	$20.00	0	4
☐ HVSLM	6' Male Serial Cable	$20.00	0	4
☐ HVSSF	1' Female Serial Cable	$20.00	0	2
☐ HVSSM	1' Male Serial Cable	$20.00	0	2

HARDWARE

CODE	PRODUCT	PRICE	LBS.	OZ.
☐ HVPRAP	VERSACARD PROM APPLE DMP	$30.00	0	3
☐ HVPRCI	VERSACARD PROM PROW/NEC/C.ITOH	$30.00	0	3
☐ HVPREP	VERSACARD PROM EPSON/PANASON.	$30.00	0	3
☐ HVPRID	VERSACARD PROM IDS	$30.00	0	3
☐ HVPROK	VERSACARD PROM OKIDATA	$30.00	0	3
☐ HZARKY	ZARDAX KEYBOARD CONNECTOR	$7.00	0	5

SOFTWARE

PEEKing at *Call-A.P.P.L.E.*

		ANTHOLOGY DISKETTES		
☐ SA1	VOL 1 (JAN-JUN 79)	$12.50	0	1
☐ SA2	VOL 2 (JUL-DEC 79)	$12.50	0	1
☐ SA3	VOL 3 (JAN-JUN 80)	$12.50	0	1
☐ SA4	VOL 4 (JUL-DEC 80)	$12.50	0	1
☐ SA5	VOL 5 (JAN-JUN 81)	$12.50	0	1
☐ SA6	VOL 6 (JUL-DEC 81)	$12.50	0	1
☐ SA7	VOL 7 (JAN-JUN 82)	$12.50	0	1
☐ SA8	VOL 8 (JUL-DEC 82)	$12.50	0	1
☐ SA9A	VOL 9A (JAN-MAR 83)	$12.50	0	1
☐ SA9B	VOL 9B (APR-JUN 83)	$12.50	0	1
☐ SA10A	VOL 10A (JUL-SEP 83)	$12.50	0	1
☐ SA10B	VOL 10B (OCT-DEC 83)	$12.50	0	1
☐ SA11A	VOL 11A (JAN-MAR 84)	$12.50	0	1
☐ SA11B	VOL 11B (APR-JUN 84)	$12.50	0	1
☐ SPA1	PASCAL ANTHOLOGY #1 (JAN-DEC 81)	$12.50	0	1
☐ SMO184	DISK OF THE MONTH 01/84	$11.00	0	1
☐ SMO284	DISK OF THE MONTH 02/84	$11.00	0	1
☐ SMO384	DISK OF THE MONTH 03/84	$11.00	0	1
☐ SMO484	DISK OF THE MONTH 04/84	$11.00	0	1

SOFTWARE

CODE	PRODUCT	PRICE	LBS.	OZ.
☐ SMO584	DISK OF THE MONTH 05/84	$11.00	0	1
☐ SMO684	DISK OF THE MONTH 06/84	$11.00	0	1
☐ SMO784	DISK OF THE MONTH 07/84	$11.00	0	1
☐ SMO884	DISK OF THE MONTH 08/84	$11.00	0	1
☐ SMO984	DISK OF THE MONTH 09/84	$11.00	0	1
☐ SM1084	DISK OF THE MONTH 10/84	$11.00	0	1
☐ SM1184	DISK OF THE MONTH 11/84	$11.00	0	1
☐ SM1284	DISK OF THE MONTH 12/84	$11.00	0	1

APPLE /// PRODUCTS

CODE	PRODUCT	PRICE	LBS.	OZ.
☐ HDR3	APPLE /// DISK DRIVE	$285.00	6	0
☐ SWW3	WORD WEAVER ///	$35.00	0	3
☐ SZARDA	ZARDAX/// (Apple ///)	$160.00	2	0

A.P.P.L.E. SERVICES

CODE	PRODUCT	PRICE	LBS.	OZ.
☐ FSOU	THE SOURCE	$60.00		
☐ FAPP	A.P.P.L.E. MEMBERSHIP ($25.00 first time application fee, $26.00 first year's dues)	$51.00		
☐ FREN	A.P.P.L.E. MEMBERSHIP RENEWAL	$26.00		
☐ FSPSVC	Special handling for rush orders	$25.00		

ORDER PRODUCTS LISTED ON REVERSE

Name _____ Membership # _____ Expiration date _____ Date _____

Address _____ City _____ State/Country _____ Zip _____ Phone _____

$3.00 Handling charge on all orders under $25.00

	CODE	PRODUCT	SHIPPING WT.	UNIT PRICE	AMOUNT
1.					
2.					
3.					
4.					
5.					
6.					

Total Shipping Weight _____ SUBTOTAL _____

Total Shipping Cost _____

Washington residents add 8.1% sales tax _____

$3.00 Handling charge if order is under $25.00 (FHDLCH) _____

TOTAL AMOUNT _____

Call TOLL FREE OUTSIDE WASHINGTON 1-800-426-3667 (Orders only — 24 hours)

PLEASE USE PRODUCT NAME AND CODE WHEN ORDERING

VISA Account No. _____ Exp. Date _____

MasterCard Account No. _____ Exp. Date _____

Mail to:
A.P.P.L.E.
21246 68th Ave. S.
Kent, WA 98032
(206) 872-2245

FOREIGN MEMBERS: Please calculate postage based on weights in product descriptions. (See postage table below for cost.)

Sorry, No Foreign Hardware Orders

FOREIGN SHIPPING CHART

POSTAL TABLE	Zone I West Indies Central America, Colombia, Venezuela	Zone II All South American Countries (except in Zone I), Europe, N. Africa	Zone III Asia, Mid East, S. Africa Australia and Pacific Islands
0 - 4 oz.	$1.28	$ 1.68	$ 2.08
4 - 8 oz.	1.80	2.60	3.40
8 - 12 oz.	2.32	3.53	4.72
12-16 oz.	2.84	4.44	6.04
16-20 oz.	3.36	5.36	7.36
20-24 oz.	3.88	6.28	8.68
24-32 oz.	4.92	8.12	11.32
2 - 3 lb.	7.00	11.80	16.60
3 - 4 lb.	9.08	15.48	21.88
Each additional ½ lb. over 4 pounds add:	1.04	1.84	2.64

CANADA SHIPPING RATES

0-4 oz.	$ .71
4-8 oz.	1.39
8-12 oz.	2.07
12-16 oz.	2.58
2 lb.	3.10
3 lb.	4.10
4 lb.	5.10

For weights over 4 lbs: $5.10 + $1.00 additional per pound.

UPS 2ND DAY AIR

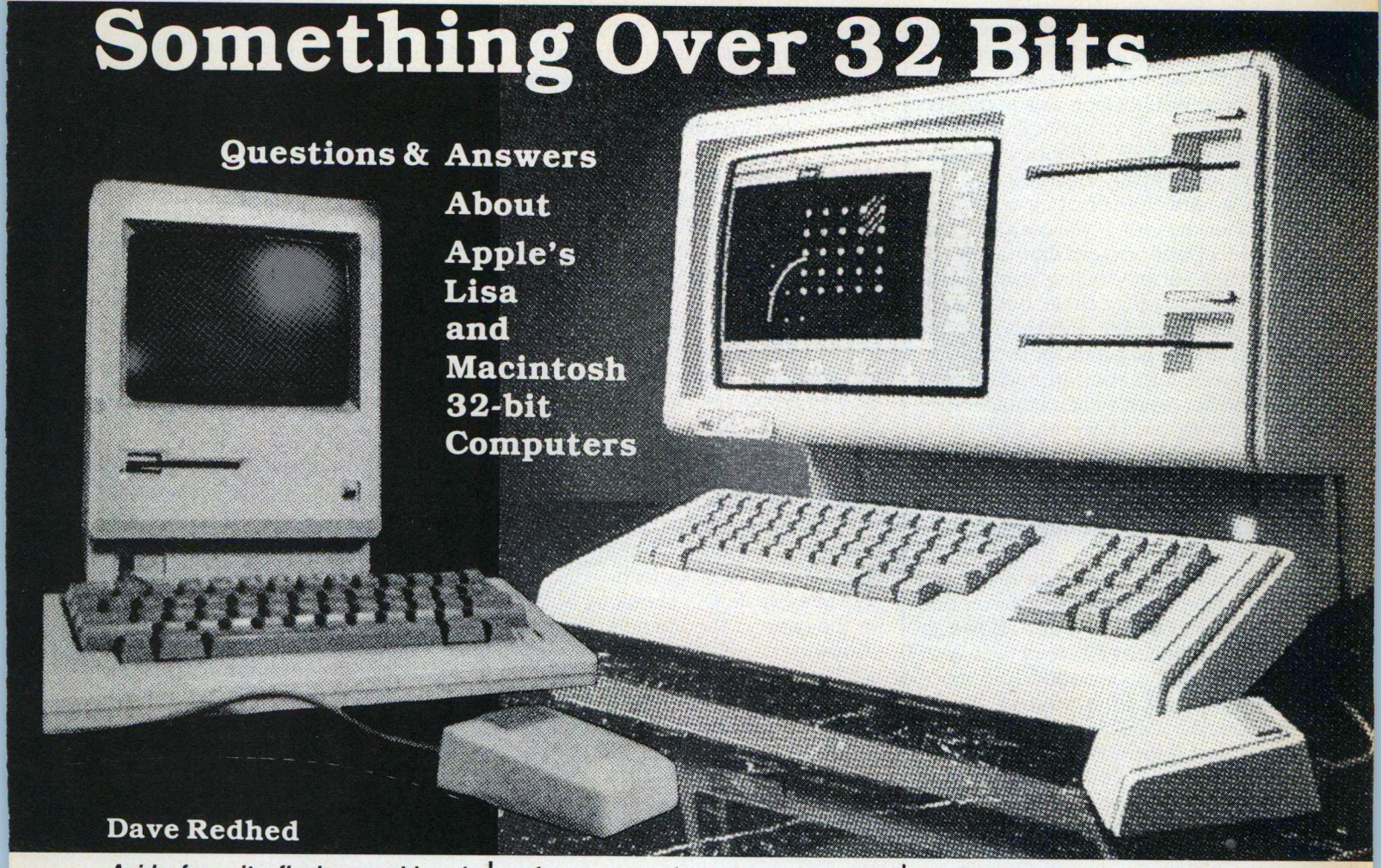
WEIGHT NOT TO EXCEED	48 STATES A	HAWAII D
1 lb.	$3.00	$4.53
2 "	4.00	5.71
3 "	5.00	6.89
4 "	6.00	8.07
5 "	6.50	9.25
6 "	7.50	10.43
7 "	8.50	11.61
8 "	9.50	12.79
9 "	10.50	13.97
10 "	11.50	15.15
11 "	12.50	16.33
12 "	13.50	17.51
13 "	14.50	18.69
14 "	15.50	19.87
15 "	16.50	21.05
16 "	17.50	22.23
17 "	18.50	23.41
18 "	19.50	24.59
19 "	20.00	25.77
20 "	21.00	26.95
21 "	22.00	28.13
22 "	23.00	29.31
23 "	24.00	30.49
24 "	25.00	31.67
25 "	26.00	32.85

PLEASE SPECIFY WHEN ORDERING:

☐ APPLE II ☐ LISA ☐ APPLE IIc
☐ APPLE IIe ☐ MACINTOSH ☐ APPLE II+
☐ APPLE III

Something Over 32 Bits

Questions & Answers About Apple's Lisa and Macintosh 32-bit Computers



Dave Redhed

Aside from its flashy graphics, is Macintosh really any different from other computers?

Yes. The Mac designers might say that there are several differences, but I believe the most significant difference is its inherent dedication to graphics. Most, if not all, other computers display character information via a character generator. Any graphics display is a separate function, both logically and electronically. The Macintosh has only a graphics mode, and everything must be drawn via software, including all text. For Apple *II* owners, the counterpart to this operation of Macintosh is the hi-res graphics mode. Some word processors on the Apple *II* use the hi-res graphics mode to produce a 70-column text display with software. This is much more flexible than the text mode on the *II*, but it is considerably slower. The Macintosh design includes a set of very fast graphics routines that can display text at the rate of over 4000 characters per second—fast enough to keep up with most typists. This single-minded focus on graphics is not found in any other personal computer on the market today.

Will we ever see a color display on Macintosh?

"Ever" can be a pretty long time (depending on your theological outlook), but I certainly would not expect to see it in the next few years. The primary roadblock is the additional memory which color requires—and memory is money. A total of 21,888 bytes of memory is dedicated to the black and white display screen (512 horizontal points by 342 vertical points). This does not count overlapping windows, which is a significant factor in a system like Macintosh. As you add colors, the memory requirements go up as a logarithmic function of the number of colors. Eight colors would require three times as much memory as black and white, and 16 colors would take 4 times as much memory. Another factor is the cost, size, and weight of a color tube—which presents some conflict with the design requirements of Mac to be small and light.

Because the underlying graphics software in the Macintosh will handle color, I would guess that you will see output onto color devices like printers and plotters long, long before you will see color on the screen. "But," you say, "if you can drive a color printer, you should be able to drive an external color terminal." Normally this would be true, but there is no provision for a color signal coming out of Mac. Color information could be sent out of Mac through one of the serial ports, but it would have to be interpreted by another processor and turned into a color signal. Color printers are already designed to take color information in the form of ASCII codes, but color monitors do not speak ASCII.

What are the compatibility problems associated with running Mac software on Lisa?

Only one serious problem exists—the shape of the individual pixels on the screen is not the same on both systems. The Mac pixels are square and come 72 to the inch. Lisa's pixels are 50% higher than they are wide (90

per inch horizontally and 60 per inch vertically). Thus a circle drawn in MacPaint on Mac will come out on Lisa as an ellipse with its long axis in the vertical direction. If you print it on Lisa it will still be a circle, but unless you wear a very special pair of glasses, you will have a hard time creating a circle in MacPaint on Lisa. If you use a program like MacDraw, you can draw along the grid lines and keep telling yourself that the ellipse is really a circle. But all is not lost; the underlying graphics routines will easily support a rather straightforward transformation in applications that bother to test which computer they are running on. However, there is no general solution that can be applied across all applications.

Is Mac a lot faster than Lisa?

At the risk of sounding sexist, the Mac's internal clock rate is faster that Lisa's by a factor of 8 to 5. However some benchmarks I have done in Microsoft BASIC do not show that much of a difference. I ran two tests involving floating point arithmetic, and the Mac times were about 85% to 90% of the Lisa times. On the surface this doesn't seem reasonable, but the figures are quite repeatable. I don't know exactly how it is handled in the Mac software, but I do know that the mouse is being continually checked for status (you can do a menu select while the program is doing calculations), and this could be a normalizing factor between the two computers.

Why do some of the font sizes on Mac come out looking so nice and others look so terrible? All the sizes I see on output from the Lisa Office System look good.

The reason they look better on Lisa is related to a megabyte of memory and a hard disk (not to mention about $2,000). Fonts take a lot of space and Mac doesn't have much to spare. Although Mac lists 9 fonts and 8 sizes, there are not 72 fonts out there on the diskette. Mac has just 22 fonts (23 on the new system diskettes) and all the rest are created by expanding or shrinking the existing fonts. The process of expanding or shrinking is done on the basis of the bitmaps, and nice shapes containing curves simply cannot be created this way. The largest font that exists in natural form is 24 point. Expanding this three times to get 72 point creates some pretty bad images. Some of the fonts have nothing smaller than 18, requiring a shrinkage of 2 to 1 to get to 9 point. Generally speaking, the shrinking process is less destructive of the original information. However, there are some really bad cases, like the 14 point San Francisco that is shrunk from 18 point. A lower case "c" is unreadable and looks like a malformed "o." Going down to 9 point turns out not to be such a terrible problem—not many people can read type that small anyway.

After their tremendous success with the Apple II, II+, and IIe, what prompted Apple to put out a computer like Macintosh with no slots and apparently very little provision to add auxiliary devices like those available for the various models of the II?

I have no inside line to the thinking of Steve Jobs or whoever it was that made this major design decision. All I know about this subject is what I read, and most of that came from the review in the February 1984 *Byte*. The overriding design constraints were to keep the Mac small and inexpensive. Making a lot of provisions for who-knows-what kinds of peripheral devices takes space and circuitry. The space drives up the size and the circuitry drives up the cost.

While I do not remember if I read about it anywhere, I think a totally separate consideration also had something to do with it. The major portion of the customer base for personal computers in 1984 and beyond (as contrasted with 1978 to 1981) is not dominated by hobbyists and computer hackers. Today most people want to buy a complete package, plug it in, and have it do whatever it is supposed to do. While we all know that computers are not usable as appliances like telephones and washers, that is certainly the direction they must take if a greater number of people are going to use computers. If taking the cover off the computer to plug boards into slots is required to make it do its function, most people will shy away from it. That is hard for the computer nuts to understand and impossible for them to sympathize with, but I believe it is an accurate observation.

We would like to encourage readers who are interested in Apple's 32 bit computers to send in questions they would like to see answered in this column. Write to : 32 Bit Questions, Call -A.P.P.L.E. Editorial Dept., 21246 68th Ave. S., Kent WA 98032.

PRINTER RIBBONS FOR APPLE COMPUTERS

NEW, FIRST QUALITY RIBBONS
FOR EPSON-C-ITOH-OKIDATA-
N.E.C. - TALLY - IDS - APPLE
MATRIX-PROWRITER AND OTHERS

•

RELOAD YOUR USED RIBBON CARTRIDGES
SAVE UP TO 50% AND MORE
CALL OR WRITE FOR MORE INFORMATION

ENCORE RIBBON SERVICES
1318 ROSS STREET, SUITE E
PETALUMA, CALIFORNIA 94952
707-762-3544
IN CALIFORNIA CALL:
1-800-431-4969

Circle 70 on Reader Service Card

Running APPLESOFT Programs with ProDOS

Richard D. Norling

LANGUAGES that become popular often show distinct variations in usage. Just as different dialects of the English language are spoken in Miami and Liverpool, so different dialects of the BASIC language are used with different computer operating systems. If a program is written in the dialect of BASIC that works with one operating system, it may need a few changes to match the dialect of BASIC that works with another operating system. That is in fact the case when Applesoft BASIC programs are moved from DOS 3.3 to ProDOS.

ProDOS, Apple Computer's new disk operating system for the Apple II+ and Apple //e, has many advantages over DOS 3.3, the old operating system. ProDOS reads floppy disk files up to eight times faster, contains an improved file directory structure, has patches to solve a few problems in the Applesoft BASIC interpreter, and introduces a great many improvements that allow new flexibility in using individual DOS commands. ProDOS is much better suited than DOS 3.3 to hard disks and other mass storage devices, and its file formats are compatible with Apple /// SOS formats.

ProDOS is an entirely new disk operating system. It was written from scratch, with the intention of maintaining as much upward compatibility with DOS 3.3 as possible. As a result, the commands issued to ProDOS from a BASIC program look quite similar to DOS 3.3 commands, but the internal workings of the two operating systems and the physical format of information on the disk are quite different.

This article describes how to make the changes in the text of your program that are necessary to make it perform under ProDOS in the same way it did under DOS 3.3. Once you get well acquainted with ProDOS, you will probably discover that some of its new command options will make it much easier to write your programs in the future. For the purpose of this article, however, we are assuming that what you want to do is to get the program running correctly under ProDOS with the smallest possible expenditure of time and effort. Therefore, this article concentrates on changing an existing program as little as possible, just enough to enable it to run efficiently under the new operating system.

Each Applesoft or DOS command will be described briefly when it is first mentioned. However, space does not permit a complete description of each command. If you want to find more detail than is provided in the descriptions given here, you can find more information in the *Applesoft II Basic Programming Reference Manual*, *The DOS Manual (DOS 3.3)*, and *Basic Programming with ProDOS*, all of which are published by Apple Computer, Inc.

What Programs Should You Convert?

If you want to use the most up-to-date operating system, you should use ProDOS for all new programs that you write. However, that does not necessarily mean that you should spend the next six months converting all your existing programs from DOS 3.3 to ProDOS. The conversion effort will be rewarded by better performance from many programs, but there's no need to insist that every program be converted.

You should consider converting any Applesoft program that uses sequential text files (the normal type of data files) in order to take advantage of ProDOS' greatly increased speed. Programs that use the DOS APPEND command to add data at the end of a text file will show the greatest improvement in speed, because ProDOS calculates the location of the end of the file, and does not have to read the entire file as DOS 3.3 did before appending the new data.

You should consider converting any Applesoft program that uses string arrays enough to cause garbage collection, because ProDOS will collect old string garbage almost instantly compared to the several minutes it often takes otherwise. If you own a hard disk, or plan to get one, you should convert everything you can to ProDOS, because the ProDOS file and catalog structure is much better suited than DOS 3.3 to hard disks. If you distribute your programs commercially, you should consider moving them to ProDOS to expand your market to include hard disk owners and those who want their files to be compatible with Apple /// formats.

You should NOT try to convert programs written in Integer BASIC unless you first translate them to Applesoft, because ProDOS does not work with Integer BASIC. The CONVERT utility program does not convert random access text files from DOS 3.3 to the ProDOS format — so if

NEW LOWER PRICES

Modula-2

Apple II $295 Apple III $395 IBM PC and XT $395 Sage II and IV $495

Volition Systems

PO BOX 1236
Del Mar, CA 92014
(619) 481-2286

Sage,™ Sage Computer Technology. Apple,™ Apple Computer, Inc. IBM,™ IBM Corp.

Circle 59 on Reader Service Card

your program uses a random access file, keep it under the old DOS unless you want to write your own program to convert random access files to the ProDOS format. There is also no way to initialize a diskette from a BASIC program with ProDOS unless you use a machine language subroutine. If your program absolutely must have the ability to initialize diskettes, then it is best left operating under DOS 3.3.

Converting your files from the DOS 3.3 format to the ProDOS disk format is as easy as pressing a few keys with Apple's CONVERT program. Just boot up the ProDOS User's Disk, and select "C" on the first menu to run the CONVERT program. Operation of the program is fully described in Chapter 7 of the "ProDOS User's Manual" that comes with the disk.

Once you have used the CONVERT program to make a version of your program in the ProDOS format, make a back-up copy so you won't lose everything if debugging of the program changes you are about to make results in destruction of disk files. Chapter 3 of the "ProDOS User's Manual" describes how to use the ProDOS FILER program to make a back-up copy of your disk. The FILER program is also on the ProDOS User's Kit disk ("F" on the first menu).

Now you are ready to change your Applesoft programs so they will work as well (or better!) with ProDOS as they did with DOS 3.3. The checklist in Figure 1 provides a quick-reference source to remind you about the kinds of changes you will need to make in each program. Each line in Figure 1 is an example of a change that is discussed in this article. The changes you may need to make in your programs are grouped into five categories: (1) Getting DOS Commands Recognized, (2) Volumes, Files, and Path Names, (3) Changes in DOS Commands, (4) Applesoft Commands to Change, and (5) Memory Management and Machine Language Subroutines.

Getting DOS Commands Recognized

Some languages, such as Apple Pascal and Microsoft's MBASIC, contain all of the file-handling commands in the language itself. When a program in one of these languages needs to access disk files or do other things normally handled by an operating system, the programmer does not even need to be aware that the language interpreter or compiler is passing the disk commands along to the operating system to be executed.

Applesoft BASIC, however, is among the languages that do not contain all of the file-handling commands in the language itself. The Applesoft programmer needs to be able to send commands to the disk operating system (or DOS) from inside the Applesoft program.

To provide a method for communication to them from Applesoft programs, both DOS 3.3 and ProDOS watch every line that gets PRINTed from an Applesoft program. If the line is recognized as a DOS command, then the command is executed instead of being printed on the screen. If the line is not recognized as a DOS command, then it is printed out just as if the disk operating system were not watching.

DOS 3.3 recognizes commands PRINTed by a BASIC program if they start with a control-D, or CHR$(4), as the first character in a line of printed output. The control-D character is not normally visible when a program is listed, so programmers often place the required control-D character in a variable named D$. Under DOS 3.3, many programmers added a carriage return in front of the control-D, as in D$ = CHR$(13) + CHR$(4), to be certain that the DOS command would start on a fresh line. Well, that won't work with ProDOS.

ProDOS, like DOS 3.3, looks for a control-D at the beginning of its commands. The ProDOS control-D must be the first character printed by an Applesoft PRINT statement, but does not need to be the first character on a fresh line. A program that starts a PRINT statement by printing CHR$(13) + CHR$(4) is printing a carriage return in front of the control-D, and the carriage return will prevent ProDOS from recognizing what that statement prints as a ProDOS command. If the program you are converting uses the CHR$(13) + CHR$(4) device to define D$ or to start DOS commands, you will need to replace that with a plain CHR$(4).

The symptom that ProDOS is not recognizing a command is simple: the command is printed out, and is not executed. If this occurs, the most likely cause of the problem is a carriage

FIGURE 1

DOS 3.3 to ProDOS Conversion Checklist

DOS 3.3	ProDOS
D$ = CHR$(13)+CHR$(4)	D$ = CHR$(4)
PRINT D$;"OPEN";:PRINT "FILE"	PRINT D$;"OPEN FILE"
PRINT D$;"OPEN FILENAME"	PRINT D$;"OPEN /VOLUME/SUBDIR/FILE"
,V 254	,V /PRODOS.VOLUME.NAME
- (no equivalent)	PREFIX /VOLUME/SUBDIRECTORY1/
- (no equivalent)	/RAM
PRINT D$;"CATALOG"	PRINT D$;"CAT"
PRINT D$;"BLOAD CHAIN,A520": CALL 520"FILE2"	PRINT D$;"CHAIN FILE2"
END	PRINT D$;"CLOSE":END
PRINT D$;"FP"	PRINT D$;"CLOSE":NEW:END
PRINT D$;"INT"	PRINT D$;"CLOSE":NEW:END
PRINT D$;"INIT ..."	- (not allowed)
PRINT D$;"MAXFILES ..."	- (ignored)
PRINT D$;"MON ..."	PRINT D$: PRINT ...: PRINT D$; "READ (or WRITE) FILE"
PRINT D$;"NOMON ..."	- (ignored)
PR# slot	PRINT D$;"PR# slot"
IN# slot	PRINT D$;"IN# slot"
I = FRE(0)	PRINT D$;"FRE"
PRINT FRE(0)	PRINT D$;"FRE": PRINT PEEK(111) - PEEK(109) + 256*(PEEK(112)-PEEK(110))
HIMEM:	(use only when no files are open and no strings are assigned; value must be an even multiple of 256)
PEEK, POKE, CALL	(check the addresses being PEEKed, POKEd, or CALLed carefully)
PRINT D$;"BLOAD FILE ..." PRINT D$;"BRUN FILE ..."	(check loading address and length carefully to avoid memory conflicts)

return, or some extra character, being printed in front of the control-D that ProDOS needs to recognize its command. The way to get the command recognized by ProDOS is to make certain that the control-D is printed as the first character of the command.

The only time the CHR$(13) was necessary under DOS 3.3 was if the last input or output operation before the DOS command was a GET, or a PRINT command that ended with a semi-colon (;), comma (,), TAB, or SPC. Those commands leave the position for the next output character on the same line as the previous input or output operation.

The best way to stay out of trouble is still to follow these two easy programming rules: (1) always PRINT what you GET, and (2) let each line of PRINT output end with a carriage return. Failure to follow these rules under ProDOS will not cause ProDOS to fail to recognize commands directed to it, but it may cause the next line

Modula-2

The Volition system recommends itself based on its superior development environment . . . its compiling speed, and its relative polish.
—Joel Pitt, PC Magazine

If you do not have access to a VAX, the next most powerful development system is a Sage with Volition's compiler; it is very fast compiling and reasonably fast running.
—Terry Anderson, Journal of Pascal, Ada & Modula-2

It is a tribute to the compactness of this programming language that it can be used on a 64K byte personal computer.
—Allen Munro, Apple Softalk

The documents that come with Volition Systems Modula-2 are about the best introduction to the language that I know of.
—Jerry Pournelle, BYTE

Volition Systems

PO BOX 1236
Del Mar, CA 92014
(619) 481-2286

printed after the ProDOS command to be printed at the end of the line that was left unfinished before the ProDOS command was issued.

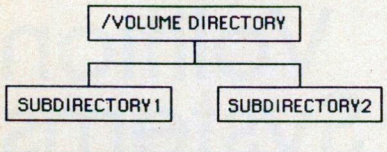
A command to DOS 3.3 could be pieced together with several PRINT statements until one of them ended without a semi-colon, and DOS 3.3 would still recognize and act on the command. Under ProDOS, the entire command must be printed by a single Applesoft PRINT statement (you can skip ahead and read the discussion of FRE now, if you can't wait to find out why). If the PRINT statement issuing your command to ProDOS ends in a semi-colon, you have a problem to correct. ProDOS will not receive the rest of the command unless you bring it into the same PRINT statement with the beginning of the ProDOS command. If the program you want to convert uses several statements to piece a DOS command together, you must do the piecing together first, and then issue the command to ProDOS in a single PRINT statement.

Volumes, Files, and Path names

Now that you are sure ProDOS will be able to recognize commands directed to it, you need to check the file names used in your program to make certain they conform to ProDOS usage. DOS 3.3, uses a single file directory on each diskette. ProDOS calls each diskette or other storage device (such as a hard disk) a *volume*, and allows layering of directories and subdirectories in a hierarchical structure such as the one shown in Figure 2. Each storage device has a Volume Directory, which can contain listings for both ordinary files and subdirectories. Each subdirectory can, once again, contain listings for both ordinary files and more directories (sub-subdirectories?).

FIGURE 2

ProDOS Directory Hierarchy



In order to access a file with ProDOS, you need to use a pathname that begins with the volume name and includes the names of all the subdirectories in the "path" to the file you want. Thus, if the file READ.ME is listed in the directory SUBDIR1 in Figure 2, then /VOLUME/SUBDIRECTORY1/READ.ME is the pathname of the file.

Pathnames would become unwieldy very fast if you had to use the whole name every time. To reduce this problem, ProDOS provides the PREFIX command. When your program issues a command to ProDOS, ProDOS automatically inserts the prefix at the beginning of the file name you provide (unless your filename begins with "/," indicating that it starts with a volume name). A normal way to use this feature in your program is to make certain the prefix is set to the correct subdirectory for that program's files, so all you have to specify in each ProDOS command is the file name itself. If you are using subdirectories and want to do this, you should use the PREFIX command in the initialization section of your program.

The hierarchical structure of directories and subdirectories can be built as deeply as you want, limited only by your disk storage space and the requirement that the pathname needed to reach each specific file not exceed 128 characters (limit of 64 in the command to ProDOS, plus limit of 64 in the prefix). As a practical matter, you will probably want to build your hierarchy only two or three levels deep so you can remember your directory structure, and so you can type a path name in something less than half an hour.

Figure 3 summarizes the changes in the rules for individual file names. File names could be up to 30 characters long with DOS 3.3, and could contain some punctuation and special characters. ProDOS file names may not exceed 15 characters, must start with a letter, and may contain only letters, numbers, and periods. This change in the rules is likely to affect a lot of programs, because the space character may no longer be included in file names. With ProDOS, directories are a special type of file, so these rules apply to directory names, too.

It is a good idea to use the CONVERT program to process both your program and all its data files into the ProDOS disk format before making changes in the program. The CONVERT program will change the file names automatically to be legal ProDOS file names. It will replace all spaces and other characters not legal in ProDOS by periods, and will discard any characters at the end of a file name that exceed the 15-character limit. After CONVERT has done its work, you should look at the catalog of your ProDOS files to learn their new names. These are the file names you should substitute for the old ones in your program.

If your program asks its user to specify a name for a file, it is good programming practice for the program to check that the name it receives is legal—no more than 15 characters, first character a letter, and no characters other than letters, digits, and periods. Lower-case letters are all right, as long as you understand that ProDOS will change them to upper case.

Volume Parameter

In many DOS commands, you are allowed to specify the Disk Volume as a safeguard to insure that the correct diskette has been inserted into the disk drive. However, use of the V parameter in DOS commands other than

FIGURE 3

Rules Governing File Names

DOS 3.3	ProDOS
Up to 30 characters long	Up to 15 characters long
Begin with a letter or one of a few special characters	Begin with a letter
All characters, including control characters, valid	Only letters, numbers, and periods valid
Spaces permitted	Spaces not permitted

INIT was comparatively rare in practice. If you didn't use it, then you have less work to do now.

The optional volume specification in DOS commands now refers to a ProDOS volume name instead of to a DOS 3.3 volume number. If your program uses the ",V" parameter in DOS calls and you want to continue to use it, you must replace the old volume number with a ProDOS volume name. A volume name for ProDOS follows the same rules as a file name, except that it must be preceded by a slash ("/"). The slash does not count as one of the fifteen characters.

/RAM

This optional change will greatly improve your program's speed if you can use it. When ProDOS is first booted, it checks to see on what model of machine it is running, and locates some of the important machine characteristics. It records the machine's configuration in the Machine Identification Byte, which is stored at location 49048 ($BF98) on the ProDOS Global Page. If the machine is a 128K Apple //e (i.e., an Apple //e with the optional Extended 80-column Card in slot 0), ProDOS sets itself up to use the extra memory as a simulated disk. It calls this simulated disk volume /RAM (pronounced "slash-ram").

You can obtain dramatic improvements in the performance of any of your programs that store things temporarily in disk files by using a /RAM file instead of a file on a much slower physical device. Because /RAM is only available on a //e with an Extended 80-column Card in slot 0, your program should test the ProDOS Machine Identification Byte to see whether or not it can use /RAM. If the value of INT (PEEK (49048) /16) is 11, then the machine is a 128K //e, and you can use /RAM. If the value is not 11, then the program should continue to use the temporary disk files it used before.

Changes in DOS Commands

Apple made a very large number of changes in DOS commands in the shift from DOS 3.3 to ProDOS. Most of those changes added new features or options to existing DOS commands, and do not require changes in the previous syntax of DOS commands. This article describes only the changes that are most pertinent to getting your Applesoft program running under ProDOS. After you have your old programs running under ProDOS, there will be plenty of time to learn how the new flexibility added to many of the other DOS commands can make your future programs easier to write.

CATALOG

ProDOS has two commands that give you a choice of a 40-column or 80-column width when you ask for a catalog of your files. If your program used the PRINT D$;"CATALOG" command under DOS 3.3, you will get an 80-column catalog under ProDOS. If you want only a 40-column catalog to match the 40-column display this command produced under DOS 3.3, you should change the command to the ProDOS short form, PRINT D$; "CAT." The contents of the catalog display will, of course, be in the new ProDOS format.

CHAIN

Chaining is a method for coping with the problem of a program too large to fit into the machine all at once. Chaining brings a new program segment into the machine and runs it without closing any files or losing the variables set by the previous program segment. The only way to chain Applesoft programs under DOS 3.3 was to execute a separate machine language program from an Applesoft program using the command sequence PRINT D$;"BLOAD CHAIN,A520": CALL 520"FILE2," where FILE2 is the name of the next program segment to be executed. Under ProDOS, that awkward method of chaining should be replaced by the simple command PRINT D$;"CHAIN FILE2".

CLOSE

When your program ends, ProDOS does not close all your files automatically like DOS 3.3 did. If you don't close all the open files yourself, ProDOS is likely to give you a FILE(S) STILL OPEN error message. Consequently, your program should clean house after it is finished running by closing all the files that it opened. This can be done quite simply with the statement PRINT D$;"CLOSE".

DOS Error Numbers

Some Applesoft programs have error processing routines that use PEEK(222) to obtain the code that indicates the type of error that has occurred. A few of the DOS error messages have been changed slightly in the shift from DOS 3.3 to ProDOS, but the causes related to most of the error codes seem to have remained the same. The principal exception is error code 3, which has changed from RANGE ERROR under DOS 3.3 to NO DEVICE CONNECTED under ProDOS. ProDOS has five more error codes (17-21) that you might want to have your program check. The messages for these codes are DIRECTORY FULL (17), FILE NOT OPEN (18), DUPLICATE FILENAME (19), FILE BUSY (20), and FILE(S) STILL OPEN (21). Appendix A of the "ProDOS User's Manual" contains a lengthy description of the ProDOS error messages.

FP and INT

These two commands disappeared in the change to ProDOS. DOS 3.3 allows you to use the FP command to switch to Applesoft and the INT command to switch to Integer BASIC. ProDOS works only with Applesoft, so it has no need for commands to switch

ULTIMA III PLAYERS

• Arm yourself with the revolutionary, high-tech, easy to use toolkit:

MagicSoft's
EXODUS CONSTRUCTION SET

• VENTURE forth into the excitement of galaxy creation with a level of power that even Lord Brittish would envy.

• Build, with your new GOD-like powers, YOUR world complete with serpents, dragons, demons... and helpless mortals.

• Place murky swamps and impregnable mountains to impede the Hero's Quest.

• Construct friendly towns and life-saving castles rife with rumors, supplies...and danger, then TRAP EXODUS on your world...thereby making Exodus vulnerable to the hero's assault.

PLAYERS OF EXODUS

• FREE! BRAND NEW! More vivid and terrifying than Sosaria: LAVALITE WORLD.

• Cross the threshold of mortal existence and enter Exodus' OWN domain...If you dare.

• All possible with an Apple][version of EXODUS: ULTIMA III, and

MagicSoft's
EXODUS CONSTRUCTION SET

All this power for a mere $28.50
Calif. residents add 6% sales tax
Send check or money order today to:

MagicSoft
529 Farragut Place
Danville, CA 94526

Circle 55 on Reader Service Card

between Applesoft and Integer BASIC. An attempt to use either of these commands under ProDOS will result in a SYNTAX ERROR message. When these commands were used from inside a program under DOS 3.3, they caused immediate termination of the program and transferred control to DOS. A similar result can be obtained under ProDOS if you replace the old DOS FP or INT command with the Applesoft sequence NEW:END — but don't forget to close all your files before executing the NEW command!

INIT

The INIT command, which formats the tracks and sectors to initialize a new disk, has been removed from ProDOS. This will prevent you from inadvertently initializing a volume, thereby destroying programs and/or data. The accepted way to initialize diskettes under ProDOS is to run the FILER program and select the initialization option. It is a good habit to initialize a whole box of new diskettes as soon as you open it, and then keep the unused ones in a known place. If your program absolutely has to initialize a diskette during execution, then you should leave it operating under DOS 3.3.

MAXFILES

DOS 3.3 had to assign buffers for its use in reading and writing disk files before your program started storing string variables, so DOS 3.3 had a command to allow you to specify how many buffers to allocate if you wanted a number other than the standard three buffers. ProDOS sets up a new buffer every time your program opens a file, and releases the space occupied by the buffer when the file is closed. Since ProDOS assigns buffer space when it is needed, it has no need for a MAXFILES command. ProDOS will ignore the MAXFILES command if you leave it in your program, but your program will be easier to understand if you remove the extraneous command.

DOS 3.3 allowed you to set a MAXFILES value as high as 16. ProDOS allows up to eight files open at the same time. If your program set a MAXFILES value higher than eight, you should check your program to determine whether it really used more than eight open files at once.

A few programs may have engaged in the forbidden practice of using the MAXFILES command to allocate file buffers and then stealing the buffers from DOS for some other purpose. If you have one of those, you will have to replace the forbidden practice with another way of allocating memory space before your program will run under ProDOS.

MON and NOMON

Here are two more commands that disappeared in the move to ProDOS. MON caused DOS 3.3 to echo information to your normal output device (i.e., your screen, unless a printer was turned on). The three MON echo options were I (input), C (commands), and O (output); all three options were commonly used at once. The NOMON command, as you probably suspect, turned the echo off.

ProDOS ignores the NOMON command, but will give you a SYNTAX ERROR message if it receives the MON command. The simplest way to deal with these two commands is to delete them from your program, but that would change the way your program operates. If you really do want to have text file input, output, and/or DOS commands echoed to the screen, the way to accomplish that under ProDOS is to insert commands to PRINT what you want to echo at the appropriate points in your program. To echo text file output, you will need to tell ProDOS to stop writing to the file with a plain PRINT D$ command before printing to the screen, and then issue another PRINT D$;"WRITE . . ." command to tell ProDOS to continue writing to the text file after you have finished printing on the screen. To echo text file input, substitute READ for WRITE in the sequence above.

Applesoft Commands to Change

FRE

This change is optional, but is highly recommended to improve program performance. The Applesoft interpreter has a way of leaving used strings lying around in the computer's memory. When no more memory is left, garbage collection is performed to free up the memory occupied by the unneeded strings. In addition to the automatic garbage collection, you can receive a special trash pick-up by using the Applesoft FRE(0) function. The Applesoft interpreter uses an inefficient algorithm for garbage collection, which can take up to 30 minutes if your program has dimensioned extremely large string arrays.

ProDOS contains an efficient garbage collection routine that should clean up all your used strings in less than five seconds. In order to ensure that its fast routine is always used, ProDOS needs a way to intercept commands before the Applesoft interpreter has a chance to start collecting garbage the old-fashioned, slow way. The ingenious method ProDOS uses is to leave the Applesoft TRACE function on all the time, and to intercept TRACE's attempt to print "#" and the line number through the zero-page output hooks when each command begins. In this way ProDOS gets a chance to see if memory is getting full so it can do its own fast garbage collection before the Applesoft interpreter has a chance. ProDOS only lets the TRACE output get printed if you have used the TRACE command. The fact that TRACE sends things through the output hooks at the beginning of every command statement is the reason you must print all of each ProDOS command in a single statement — otherwise the command to ProDOS will get interrupted by the TRACE output and go unrecognized.

If your program prints the free memory value that is returned by the FRE(0) function, or uses the value in a calculation, you can still get the benefit of ProDOS' speed. Just tell ProDOS to do the garbage collection with PRINT D$;"FRE," and then get the free memory value with PEEK(111)-PEEK(109) + 256*(PEEK(112)-PEEK(110)).

PR# and IN#

Both Applesoft and DOS have these two commands in their vocabularies. Use of the Applesoft versions of these two commands was supposed to be forbidden even under DOS 3.3, but the Applesoft version may still have been used on occasion. If the Applesoft version of either of these commands is used under ProDOS, your program will fail to function properly. ProDOS will be disconnected, and will not be able to receive its commands. In addition, the TRACE function that is kept on by ProDOS may be left on

when ProDOS is disconnected, resulting in display of unwanted TRACE output on your screen or printer. Replace the Applesoft version of these commands with the ProDOS commands PRINT D$;"PR#slot" and PRINT D$;"IN#slot".

Memory Management and Machine Language Subroutines

Conversion of machine language programs from DOS 3.3 to ProDOS is a complicated subject that could easily justify a separate article or even a short book. However, a brief summary will be included here because many Applesoft programs make limited use of machine language subroutines. If you use machine language routines extensively, you should obtain and read the "ProDOS Technical Reference Manual" published by Apple Computer. That manual is an excellent source of technical information about ProDOS and will be almost indispensable for serious machine language programmers.

Memory Allocation

ProDOS contains its own memory manager, and moves things around in your machine's memory a lot more than DOS 3.3 did. As a result, it is a bit harder for you to section off a piece of memory and reserve it for your special purpose. Apple warns that the only safe places for you to put things are in the familiar neighborhood between $0300 and $03CF (768 and 975 for those who think in decimal) and above HIMEM (using a special ProDOS procedure). ProDOS keeps track of its memory allocations in a special bit map, and will not allow you to load files on top of areas that ProDOS has already reserved.

HIMEM:

The pointer to the highest memory location available to your program's variable storage is called HIMEM. It can be changed by using the Applesoft HIMEM: command. Changing HIMEM in your program has always been a bit dangerous unless you were very sure you knew what you were doing, but under ProDOS it can be disastrous.

Unlike DOS 3.3, ProDOS changes the HIMEM value every time your program opens or closes files. If your program changes the HIMEM value, you run a substantial risk of either storing something in an area that ProDOS thinks is an open file buffer, or of having ProDOS create a file buffer on top of whatever you tried to hide above HIMEM. It is not safe to tamper with HIMEM if your program has any text files open.

Moving HIMEM upwards could cause destruction of file buffers or part of ProDOS itself. Moving HIMEM downwards usually results in destruction of the contents of string variables set by the Applesoft program. If you must move HIMEM downwards, you should still comply with the ProDOS rule that HIMEM must always point to a page boundary—an address such as $7800 or $9D00 that ends in two zeros when written in hexadecimal notation. In decimal arithmetic, you can identify a page boundary by the fact that it can be divided by 256 with no fraction left over afterwards.

The officially-sanctioned method for hiding machine language programs or other things above HIMEM is quite different from the DOS 3.3 procedure. A full description of its use would be too long for this article. The procedure is described briefly in Appendix A of the "ProDOS Technical Reference Manual".

PEEKs, POKEs, and CALLs

These commands provide a way for Applesoft programs to work directly with machine language. PEEK is used to obtain the value stored in a specified memory location, POKE is used to put a value into a specified memory location, and CALL transfers control to a machine language subroutine that starts at the specified memory location. The POKE command is sometimes used to put a short machine language program directly into memory from an Applesoft program instead of BLOADing it from a separate disk binary file.

Figure 4 lists some of the key locations that might have been PEEKed, POKEd or CALLed from an Applesoft program running under DOS 3.3 that will need to be changed before your program will work correctly under ProDOS. They fall into three areas. The first area is the input buffer on page 2 of memory ($0200–02FF). Page 2 was used occasionally under DOS 3.3 for machine language programs that were just a little bit too big for the $0300–03CF area. However, ProDOS makes use of page 2 much more than DOS 3.3 did. It is no longer safe for your program to use page 2.

FIGURE 4

Addresses to Avoid in PEEKS, POKES, and CALLS

$0200-02FF	512- 767	Input buffer
$03D6-03EF	982- 1007	DOS 3.3 vectors
$9D00-BFFF	40192-49151	48K DOS 3.3

Since ProDOS is a completely new program, it should not be surprising that any PEEKs, POKEs, and CALLs into DOS 3.3 or the vectors it established on memory page 3 will not work the same with ProDOS. Because the DOS 3.3 warm start and cold start vectors at locations $03D0 and $03D3 were used so extensively, the authors of ProDOS replaced them with similar vectors for ProDOS. But the other DOS vectors that were located from $03D6 to $03EF are gone, and were not replaced.

BYTES & PIECES PRICE RIOT — Ask For Free Price List

$1.00 credit for phone orders over $100.00

	LIST	SPECIAL
Bookends	125.00	90.46
Chess (Odesta)	69.95	50.76
Crossword Magic	49.95	35.96
Ess. Data Dup. (EDD)	79.95	58.16
Flight Simulator II	49.95	38.96
G.P.L.E.	49.95	35.96
Gibson Light Pen	249.95	196.96
Graphics Department	124.95	90.46
Graphics Magician	59.95	43.36
Homeword	69.95	50.76
Inspector	59.95	43.36
Jane	295.00	215.25
Koala Pad	124.95	90.46
Locksmith 5.0	99.95	77.96
Maxell Disks (10)	51.00	29.00
Phi Beta Filer	49.95	35.96
Questron	49.95	35.96
SAT (Harcourt Brace)	79.95	58.16
SAT (Krell)	299.95	252.96
Sensible Speller	125.00	90.46
16 K Card (Microsoft)	99.95	75.46
Think Tank (II+ or IIe)	150.00	108.96
Time is Money	99.95	71.96
Trains	39.95	28.56
Wildcard 2	139.95	110.19

Ask for our educational software catalog

C.O.D. • M.O. • Cert. Checks • School P.O. • VISA • MasterCard • Checks Allow 2 Weeks • N.Y.S. Res. Add Sales Tax • US Orders Under $150. Add $2.00 P & H • All Canadian, US Funds $3.00 P & H. • Foreign, Charges Only, Min. P & H $7.00

Bytes & Pieces, Box 525, Dept. C, East Setauket, N.Y. 11733, (516) 751-2535

Circle 109 on Reader Service Card

The contents of these locations are now unpredictable, and a CALL to one of them will probably cause your program to crash into the monitor.

The locations ($9D00–$BFFF) shown for DOS 3.3 in Figure 4 are the locations that were occupied by a normal DOS 3.3 in a machine with at least 48K of RAM. You should apply the appropriate adjustment if yours is one of those increasingly rare machines that has less than 48K. The area formerly occupied by DOS 3.3 is now occupied by ProDOS, and none of the locations have the same purpose. The values returned by PEEKs will be meaningless. The effects of a POKE or CALL into one of these locations could cause the ProDOS system to crash. All PEEKs, POKEs, or CALLs to locations from 40192 through 49151 should be removed from your program, even if that requires you to make some changes in the way your program operates.

BLOAD and BRUN Addresses

When your program BLOADs or BRUNs a binary file, it can specify in the DOS command the address at which the file is to be loaded. If the address is not specified in the DOS command, the address from which the file was originally BSAVEd will be used. To avoid conflict with ProDOS over memory allocation, you should check the load addresses and lengths of all programs that will be BLOADed or BRUN from your program to determine what area of memory the programs will occupy when they are loaded.

One of the advantages of ProDOS over DOS 3.3 is that the default address and length of each binary file is displayed when you use the PRINT D$;"CATALOG" command to get an 80-column display of the directory. If a BLOAD or BRUN address is not specified in the command in your Applesoft program, you can find the address along with the length by looking at the catalog. Add the file length to the loading address, and then subtract 1 to get the last address that the file will occupy. If any part of the file would be loaded on top of ProDOS ($9D00 and up), the low-resolution screen ($0400–$07FF), the zero page and stack area ($0000–01FF), or the ProDOS file buffers for any files you have open, your program will receive a NO BUFFERS AVAILABLE error message from ProDOS when it tries to execute the BLOAD or BRUN command. You must take responsibility for ensuring that no part of the file will be loaded on top of your own program and variables, or on top of the remaining vectors on page 3 ($03D0–$03D5 and $03F0–$03FF).

Conclusions

WARNING: Only a few machine language programs are written in truly relocatable code. If you need to change the loading address of a machine language program, you probably need also to reassemble the program to run at the new location. If the machine language program had any interactions with DOS 3.3, those portions of the program will need to be changed to use the proper ProDOS Machine Language Interface. That interface is described in Apple Computer's "ProDOS Technical Reference Manual".

This article has described how to make the types of changes necessary to convert an Applesoft program that now runs with DOS 3.3 so that it will run with ProDOS. The list of changes you might have to make is relatively long, but in practice many programs that do not use data files will require very few changes, or even no changes at all. Don't let the length of the list in Figure 1 scare you. Most of the commands that need to be changed in the transition from DOS 3.3 to ProDOS are not used very frequently. They were included here to prepare you should you happen to encounter one of them. Even if some changes will be necessary for your special program, I hope you will consider converting it to obtain the many benefits of ProDOS.

The list of changes included in this article is necessarily somewhat preliminary, because people have just begun converting the vast array of available Applesoft programs from DOS 3.3 to ProDOS. Every effort has been made to try to make this list complete, but it is quite possible that additional conversion difficulties remain to be discovered. If you happen to find anything I have overlooked, I hope you will contact me about it through *Call -A.P.P.L.E.*.

Laing Electronics

μ-SCI
MICRO-SCI
A2 DRIVE

$195

BUY THE FIRST AND THE BEST APPLE COMPATIBLE DRIVE!

A2 DRIVE WITH CONTROLLER $309
(FREE SHIPPING IN USA OR CANADA. A2 MODELS ONLY.)

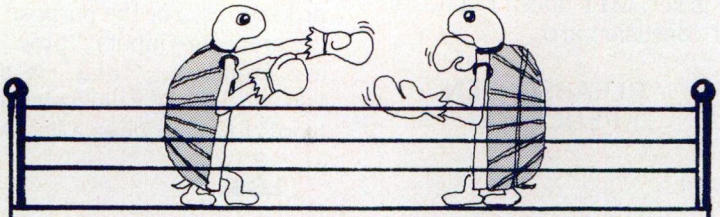
A40 (40 TRACK DRIVE W/O CONTROLLER) $284
A70 (70 TRACK DRIVE W/O CONTROLLER) $365
— CONTROLLER FOR A40 OR A70 DRIVES $ 79
A3 APPLE III DIRECT REPLACEMENT $299

10382 Stanford Ave.,E-203, Garden Grove, Cal., 92640
Phone (714) 534-4216

Check or Money Order. — NO CREDIT CARDS! Check's Allow 10 BANKING DAYS FOR CLEARANCE. Prices and specifications subject to change without notice. Add $7.50 for UPS surface. Cal. Res. add 6% Sales Tax.

Circle 116 on Reader Service Card

Logo vs. Logo



Apple users who desire logo have a choice between two variants of this increasingly popular language. How significant are the differences between them?

Jeff Sandys

THERE ARE two major dialects of Logo. The version sold under the names Terrapin and Krell was developed at the Massachusetts Institute of Technology (MIT) by Harold Abelson, the author of *Turtle Geometry*. Apple Logo, on the other hand, was developed by Logo Computer Systems Inc. (LCSI), under the guidance of Seymour Papert, the inventor of Logo. Both versions are well-thought-out implementations of Logo.

Most of the differences between LCSI and MIT Logo are matters of style. I feel that the LCSI version is slightly more similar to LISP, while the MIT version is a little more like BASIC. The MIT version also tends to be a bit more oriented toward children than the LCSI version. The translation guide provided here will illustrate the differences, and will help you convert Logo programs from one version to another. It might also help potential buyers compare the features while deciding which one to get.

In this comparison, "LCSI" refers to Apple Logo and "MIT" means Terrapin Logo Version 2.0. This is not an absolute translation, as there are subtle differences in the way some of the primitives operate. The following words have the same meaning in both languages:

ASCII	ITEM
BK or BACK	LAST
BF or BUTFIRST	LT or LEFT
BL or BUTLAST	LIST
CATALOG	LOCAL
CHAR	LPUT
CLEARTEXT	MAKE
CO	NOT
COS	OP or OUTPUT
COUNT	PADDLE
DEFINE	PAUSE
ERASEFILE	PD or PENDOWN
FIRST	PU or PENUP
FD or FORWARD	PR or PRINT
FPUT	QUOTIENT
FULLSCREEN	RANDOM
GO	RC
HEADING	REMAINDER
HT or HIDETURTLE	REPEAT
HOME	RT or RIGHT
ROUND	STARTUP
RUN	STOP
SAVE	TEST
ST or SHOWTURTLE	TEXT
SE or SENTENCE	THING
SETDISK	WORD
SETH or SETHEADING	WRAP
SETX	XCOR
SETY	YCOR
SIN	.BPT
SPLITSCREEN	.DEPOSIT
SQRT	.EXAMINE

Like LISP, LCSI uses words ending in "P" as predicates, while MIT uses "?." A predicate is a function that outputs TRUE or FALSE. The following primitives have the same meaning but use different words. The LCSI word is on the left and the equivalent MIT word is on the right:

LCSI	MIT
AND	ALLOF
BUTTONP	PADDLEBUTTON
EMPTYP	EMPTY?
EQUALP	EQUAL?
ERN	ERNAME
FENCE	NOWRAP
INT	INTEGER
KEYP	RC?
LISTP	LIST?
LOAD	READ
MEMBERP	MEMBER?
NAMEP	THING?
NODES	.NODES
NUMBERP	NUMBER?
OR	ANYOF
READCHAR	READCHARACTER
RL or READLIST	REQUEST
RECYCLE	.GCOLL
RERANDOM	RANDOMIZE
SETSCRUNCH	.ASPECT
TYPE	PRINT1
WORDP	WORD?
.PRINTER	OUTDEV

While programmers will know the meaning of OR, children may prefer ANYOF because it reads better. Fortunately you can easily add new procedures to both languages. If you want SAME to mean EQUALP or EQUAL?, so be it. There are four words with the same name but with different meanings. LCSI uses the prefix "SET" when a parameter is set, MIT doesn't. The four words that cause the most confusion are:

BACKGROUND	CLEARSCREEN
CURSOR	PENCOLOR

From MIT to LCSI use:

BG or BACKGROUND	SETBG
CS or CLEARSCREEN	CLEAN
CURSOR	SETCURSOR
PC or PENCOLOR	SETPC

From LCSI to MIT substitute these phrases. (The CURSOR.HV procedure is found on the utility disk.)

BG or BACKGROUND	LAST BL TURTLESTATE
CS or CLEARSCREEN	DRAW
CURSOR	CURSOR.HV
PC or PENCOLOR	LAST TURTLESTATE

The following two lists include examples of how MIT Logo uses extra words, often called *noise words*, to enhance readability. For example, in MIT the words THEN and ELSE have no meaning outside of an IF statement. Also, the words ALL, NAMES, and PROCEDURES can only be used with ED, EDIT, ER, ERASE, and PO.

In the first group below, notice how LCSI relies on the list structure of Logo rather than noise words. The use of quotes, dots and brackets are shown as required by the procedures. For the purpose below, words in small letters are procedure inputs; "proc" is a procedure name, x and y are numbers, "cond" is a true or false expression, "do . . ." is any valid instruction, and "word" is any valid word. LCSI requires a list, MIT doesn't:

LCSI	MIT
ARCTAN x	ATAN x 1
ARCTAN (x/y)	ATAN x y
IF cond [do.true]	IF cond THEN do.true
IF cond [do.t] [do.f]	IF cond THEN do.t ELSE do.f
IFF [do.false]	IFF do.false
IFFALSE [do.false]	IFFALSE do.false
IFT [do.true]	IFT do.true
IFTRUE [do.true]	IFTRUE do.true
LABEL "word	word:
SETPOS [x y]	SETXY x y
TOWARDS [x y]	TOWARDS x

As in LISP, LCSI "evaluates" input data and variables. That is, if the input itself is a procedure name, then LCSI immediately executes it. MIT doesn't. The list below shows the equivalent commands for EDIT, ERASE, and PO (output).

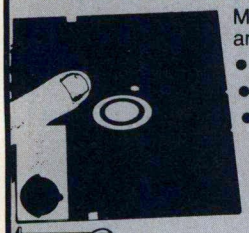
LCSI	MIT
EDNS	ED NAMES
ED or EDIT "proc	ED or EDIT proc
ER or ERASE "proc	ER or ERASE proc
ERALL	ER ALL
ERNS	ER NAMES
ERPS	ER PROCEDURES
PO "proc	PO proc
POALL	PO ALL
PONS	PO NAMES
POPS	PO PROCEDURES

The word TO is used to define a new procedure. In MIT it clears the text and hi-res screens and activates the editor. In LCSI the text and hi-res screens are not erased, and remain after a new procedure is defined.

The math symbols can be used as infix operations in both languages. LCSI also allows math symbols to be used as prefix operations. That is the instruction **PRINT + 2 3** is the same as **PRINT 2 + 3** in LCSI but would not work in MIT. Unary minus (or negation) is done differently in the two dialects. In LCSI the minus sign is attached to the number like this: -2. In MIT the negative number must be inside parentheses like this: (- 2). Of course you can put parentheses around negative numbers in LCSI Logo. The math symbols are:

+ - * / < = >

***REDUCES DISKETTE COST 50%!**

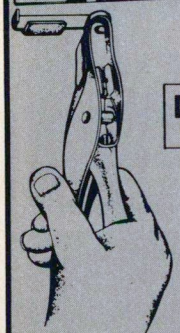


Make use of the back of your 5¼" Diskettes and SAVE

- ***NIBBLE NOTCH*** tools make it easy.
- Adds the precise notch where you need it.
- Doubles diskette space or MONEY BACK!

NIBBLE NOTCH I
Cuts square notch for Apple, II, II+, IIe, III, Franklin and Commodore.
only **$14.95** each
add $1.50 each P&H ($4.50 each foreign P&H)

DISK OPTIMIZER© SYSTEM
SOFTWARE FOR APPLE, II, II+, IIe, III & Franklin



- **469% FASTER Than Similar Programs!**
- Certifies your "new" disk 100% Error Free
- Removes Bad Sectors • Adds 36th Track
- Performs Disk Drive Speed Check
- Adds DOS and More

only **$24.95** each
add $1.50 each P&H ($4.50 each foreign P&H)

— OR BUY BOTH —
only $29.95
add $2.50 each set P&H ($6.50 foreign P&H)
Florida Residents Add 5% Sales Tax

SATISFACTION GUARANTEED OR YOUR MONEY BACK!

ORDER TODAY!
VISA MasterCard
CALL TOLL FREE: 1-800-642-2536
OR SEND CHECK OR MONEY ORDER TO:
NIBBLE NOTCH COMPUTER PRODUCTS
DIVISION OF CORTRAN INTERNATIONAL
4211 NW 75th TERRACE • DEPT. 7 6 • LAUDERHILL, FL 33319
PAT. PEND. ALL TRADEMARKS ARE ACKNOWLEDGED

Circle 57 on Reader Service Card

Many of the words in one dialect can be converted to an equivalent phrase in the other dialect. For converting LCSI to MIT use:

LCSI	MIT
PX or PENREVERSE	PC 6
PE or PENERASE	PC LAST BL TURTLESTATE
PRIMITIVEP "proc	EQUAL? "proc TEXT "proc
DEFINEDP "proc	NOT EQUAL? [ ] TEXT "proc
SHOWNP	FIRST BF TURTLESTATE
POS	SE XCOR YCOR

Procedures for the LCSI words DOT, PEN, SETPEN, SHOW and WAIT can be written in MIT but are too long to present here.

To convert MIT to LCSI use:

MIT	LCSI
CLEARINPUT	.DEPOSIT 846 .EXAMINE 847
CONTINUE	CO
TOPLEVEL	THROW "TOPLEVEL
GOODBYE	.SYSTEM
;	REM

The words for .SYSTEM, REM, TRACE and NOTRACE can be found on the toolkit disk. Definitions for the MIT words .CALL and TURTLESTATE are too long to present here.

Significant Differences

Now we come to the unique features in both languages. There may be ways to define these words in the other dialect but I have not figured out a way yet. The unique feature of MIT Logo is access to DOS 3.3. You can save and load pictures and enter most DOS commands. The utility disk has procedures to read and write text files. Also MIT's turtle is an Applesoft shape and can be redefined. These words are available in MIT but not in LCSI:

DOS, ERASEPICT, READPICT, SAVEPICT

To save pictures and use DOS commands with LCSI, boot a DOS disk from Logo with .PRINTER 6. This kills Logo and starts Applesoft without erasing the hi-res screen.

LISP programmers will recognize many of LCSI's unique functions. LCSI Logo has property lists, error handling and some extra features. Property lists are something like arrays and could be implemented in MIT but it would be very inefficient. The following words are found in LCSI Logo but are not available in MIT Logo:

BURY
CATCH
COPYDEF
DISK
ERRACT
ERROR
GPROP
PACKAGE
PKGALL
PLIST
PPROP
PPS
PRODUCT
REDEFP
REMPROP
REPARSE
SCRUNCH
SUM
THROW
UNBURY
WINDOW

There are also differences in the internal structure of the two Logos. Nodespace in LCSI extends from $4000 to $910D for 4149 five-byte nodes. However LCSI keeps the primitive names and stack in nodespace, and only 2825 nodes are available to the user. In MIT nodespace is on the RAM card from $D000 to $FFF7 for 2456 nodes. The primitive names and stack are outside of nodespace so the programmer has 2273 nodes available. Also MIT compacts some user-written procedures to save space. In LCSI, procedures are always stored as lists.

The two Logos are about the same speed. Benchmarks show that LCSI is a little faster with graphics procedures, but for list processing it is a dead heat. The Sieve of Eratosthenes, *Byte* magazine's benchmark, ran faster with MIT. MIT also loads files faster, as LCSI uses text files.

The most noticeable difference is in booting. LCSI boots in 10 seconds, while MIT takes 40 seconds.

Conclusions

As an overall summmary of the differences in style between LCSI and MIT Logo, it may be fair to say that LCSI is designed for precision and functionality, while MIT—with its somewhat less formal syntax—is designed for easier readability. Both versions of Logo will give may hours of programming pleasure. Most of the differences are easy to overcome. Both languages have clear error messages to guide you past mistakes. The editors and the major primitives work the same way. The choice between the two Logos may come down to DOS commands or property list and error handling. Whatever you choose, you can't go wrong by learning Logo.

"TELEDELIVERY"?

- SOFTWARE ON DEMAND
- FASTEST DELIVERY ANYWHERE
- 24-HOUR ACCESS, WORLDWIDE
- SAVE TIME, TRAVEL, MONEY
- EXCELLENT CUSTOMER SUPPORT

TRY IT NOW!

CALL, VIA MODEM: (213) 516-9432
AND EXPERIENCE THE FUTURE.

TELEPHONE SOFTWARE CONNECTION® INC

POST OFFICE BOX 6548 - TORRANCE, CALIFORNIA 90504

Circle 6 on Reader Service Card

A.P.P.L.E. Doctor

Bruce F. Field

APPLEWRITER II UNDERLINING AND THE OKIDATA

LAST MONTH I answered a question about underlining with Applewriter II and an Okidata printer. The problem was that if the printer underlining feature was turned on, and the words to be underlined continued from the right side of one line to the next line, the underline continued through the left margin. The solution is to use the backslash command in Applewriter rather than sending the "underline on" command to the printer. Right after sending the column in for publication I received several more letters describing exactly the same problem. My solution still stands for all printers except the Okidata. Unfortunately the Okidata refuses to recognize the backspace character which is part of what Applewriter uses to underline when the backslash command is given. If the left margin command of Applewriter is used (i.e. .lmXX), when Applewriter reaches the end of a line it sends XX spaces at the beginning of the next line. If the backslash command of Applewriter is used, Applewriter is smart enough to know not to underline the spaces. On the other hand, if the printer underline feature is turned on, the spaces look like any other character to the printer and are underlined. If you can't use the backslash command (as with the Okidata), another solution is to always use a left margin of 0 with Applewriter (.lm0) and instead send the proper command to the printer to set the left margin as desired. For the Okidata printer the left margin command is <ESC>%Cnnn, where nnn is the number of dot columns for the left margin. For a left margin equivalent to about 10 characters, nnn should be 080. Leading zeros are required.

FASTEN YOUR KEYBOARD

I have received a letter from Robert Knight of Guilderland, New York, regarding his experience with keyboard problems, that is worth passing along.

". . . (I have had a keyboard problem), specifically either multi-printing of a letter on a single keypress or no action at all on a keypress.

I too have been to the dealer on three occasions for this problem and have been charged for one decoder board, the repair of a decoder board, and finally a recheck at no cost.

I took the bull by the horns and took the keyboard off when I found that reseating the connector appeared to solve the problem temporarily. Cleaning the connector contacts seemed to improve the condition for awhile but did not solve the problem.

I checked every solder connection on the printed circuit side of the keyboard and I found three broken connections—two on keys and one on the wires that lie on the board. I soldered these connections about six months ago and have not had any problems since.

Before paying for a decoder, I would recommend getting a 10X magnifying lens and carefully check those solder joints."

PERILS OF PASCAL

Question

I use the PASCAL language with my Apple II+ and EPSON MX-100 printer. I have a CCS 7710B asynchronous serial interface but cannot print from PASCAL. Can you advise me?

Answer

Pascal should automatically recognize the CCS card as a serial interface if it is plugged into slots 1, 2, or 3. To be used for the printer output function of PASCAL the card must be in slot 1. You should have no problem if this is the case. In the PASCAL system, slot 2 is reserved for a modem, and slot 3 is used for a terminal (or an 80-column card).

Question

My problem concerns the PASCAL error #253, "procedure too long." When I write a program in PASCAL everything is fine until it gets long, then I get the "procedure too long" error even if there is no procedure in it. When I get the error I can eliminate it by reducing the size of the program. Often I can make it run just be taking out 1 or 2 characters. I have seen programs many times longer than mine so the real problem isn't length. Any help you can give me with this problem would be appreciated.

Answer

Apple PASCAL has an internal limitation of 1200 words (2400 bytes) of object code per procedure. Your entire program is a procedure and is causing the "too long procedure" error. The solution is to break up your program into a number of smaller procedures. You can have a lot of small procedures as long as each procedure is smaller than 1200 words of object code. (Object code is what the system produces on compilation of the source code that you type in.)

CLIPPED CABLES

Question

I have the new Macintosh computer and Imagewriter printer. The printer works fine when I use the high resolution character printing mode from MacWrite; but any other combination, such as using MacPaint, produces garbage on the printer.

Answer

Some early model printers were shipped with a defective cable. Pin 20 was clipped off on the printer end with the expectation that the XON/XOFF protocol, rather than DTR, would be used with the printer. See your Apple dealer to get a replacement cable.

PLUGGING IN THE PLUS

Question

I have an Apple II (Integer BASIC in ROM), and a 16K card, generally used to contain Floating-point BASIC. I've noticed that several programs on the market specify that one needs an Apple II+ in order to run their software. More importantly, ProDOS does not support Integer BASIC. What alternatives do I have? For instance, can I purchase the Floating-point ROM set, and simply substitute them for the Integer ROMs? Or, can I purchase a Floating-point Firmware card and still be able to use my 16K card? And, who sells it?

Answer

You can plug the Floating-point BASIC (Applesoft) ROMs right in the motherboard of your II without any other modifications and have the functional equivalent of an Apple II+. If you purchase a firmware card I still recommend that you put the Applesoft ROMs on the motherboard, and if you wish you can put the Integer BASIC ROMs on the card. And yes, you can use both the firmware card and the 16K card. The best way to do this is to put the 16K card in slot 0 and put the firmware card in another slot. Slot 4 is a good choice because the switch on the back of the card can fit through the slot in the case. For more information on using the firmware card see Dave Morganstein's article "Using Firmware Card in Slot 4" which appeared in the November 1981 issue of the *Washington Apple Pi* journal and the May 1982 issue of *Call -A.P.P.L.E.* Since the Applesoft ROMs are copyrighted you will have to get them from an authorized Apple dealer.

EITHER—OR

Question

Should the disk drive door be left open or closed when not in use?

Answer

I am sure that no matter what I say someone out there will take the opposite viewpoint. Seriously though, if you leave your computer on all the time and leave a disk in the drive with the door closed, then if there is a power failure when the computer comes back on it will boot DOS instead of turning on the drive and running forever. On the other hand, if you are referring to closing the door when the computer is turned off, I am of the opinion that in a reasonably clean environment the disk drive door should be left open. Closing the door with no disk in the drive lets the pressure pad, which is coated with oxide residue and other garbage, come in contact with the read/write head.

APPLE III 5-BIT SERIAL COMMUNICATION

Question

I am trying to interface an Apple III to the Telex I Service. (Telex I assumes that the terminal is a Teletype using 5 hole punch tape.) I have written the necessary software to communicate with Telex I in its "5-level Code," but I am unable to get the Apple III's RS232 Driver to send 5 data bits or find a modem that can work with 5 data bits. Any suggestions?

Answer

First lets deal with the modem problem. The electrical signals generated by a computer are not compatible with the telephone lines. A modem is simply a circuit to convert the computer signals into tones that can be sent over the telephone. The number of data bits, stop bits, or whatever are of no concern to the modem. You do have to worry about the speed with which the data is sent, however Teletypes communicate at (yawn) 110 baud so that any 300 baud modem (the most popular kind) is more than fast enough for you.

Your problem boils down to not being able to get the Apple *III* to output 5 data bits and this is a limitation of the Apple *III* RS232 Driver. One way around this would be to buy an RS232 serial card capable of 5 bit operation but as far as I know such a card doesn't exist. Another solution is to talk to the Bizcomp modem people; they have a software program written for the II that uses the game I/O port to drive the modem and does not require any serial card. It might be possible to modify this for use with the *III*.

STRETCHING BINARY FILE

Question

In regard to the BSAVE command, the length parameter can not exceed 32767. If a file exceeds this length it must be split into 2 files and then saved separately. Is there any way to get around this restriction so the file can be saved under a single name?

Answer

Sure, change memory location $A964 in DOS from $7F to $FF. From Applesoft this can be done with POKE 43364,255. This changes the range attribute table in DOS to allow binary files as large as 65535 bytes.

DOUBLE HI-RES ON THE II+

Question

Why can't I do double hi-res graphics on my II+. From all I have seen it requires extra memory, but not 64K of it. Is there some way to superimpose both HGR and HGR2 screens at once or prevent the Apple from skipping every other line?

Answer

In addition to the extra memory in the *IIe* to do double hi-res, the graphics circuits have been modified. There are actually two hi-res 1 screens and two hi-res 2 screens in the *IIe* (with the 64K extended memory) and both hi-res 1 (or hi-res 2) screens can be displayed simultaneously to produce 560 X 192 resolution graphics. In your II+ the circuits are not designed to display both hi-res screens at the same time—it's one or the other.

However, some interesting double hi-res effects can be done on the II+ if you realize that turning on the color-shift-bit (the eighth bit) shifts the dots on a monochrome monitor a half-bit position to the right. If you were to plot on hi-res screen 1 with HCOLOR=3 (or 1 or 2) and plot on hi-res screen 2 with HCOLOR=7 (or 5 or 6) and then rapidly switch the screen being displayed on the monitor back and forth between 1 and 2 (this would have to be done from machine language) you would have 560 X 192 resolution.

1-800-356-7511
TOLL FREE
OVER 3,000 ITEMS CARRIED
ALL COMPETITIVELY PRICED!
WISCONSIN CALL 608-868-4777
CALL FOR EDUCATIONAL SPECIALS!

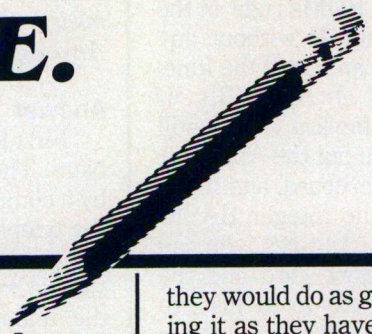
Barrons S.A.T.	68.95
d Base II	319.95
C. Itoh ProWriter (8510)	389.95
C. Itoh Ribbons	5.95
Dollars and Sense	68.95
Format II Enhanced	109.95
4th Dimension Super Drive II	229.95
Friday!	199.95
Helix (MacIntosh)	289.95
Locksmith 5.0	78.95
Micromodem IIe w/Smartcom	239.95
Multiplan	134.95
Rocky's Boots	38.95
Star Gemini 10X Printer	299.95
Sticky Bear Series	29.95
Word Star	289.95

IN-STOCK ITEMS SHIPPED WITHIN 24 HOURS. Free catalogs available. Shipping add $2.50. Visa/Master-card add 4%. Wis. residents add 5% sales tax.

name•brand software
P.O. Box 206
Milton, WI 53563

Circle 62 on Reader Service Card

Write—A.P.P.L.E.



A.P.P.L.E. Kudos

I read all the magazines that write about Apples and this one is my favorite. I can't think of a better source of useful programs, tips, information, and tutorials for an interested user of Apple computers than this magazine.

The back issues are a wonderful way for beginning users to get acquainted with their new machine. More advanced users will appreciate the clearly written articles by writers who articulate their own interests with knowledge and enthusiasm.

There is a lot more to A.P.P.L.E. than the magazine: the software library; the inexpensive hardware; the compendiums of knowledge about Applesoft, Pascal, and DOS; my fellow members who have made their knowledge available to the rest of us; and a feeling of community. But let's not forget about the magazine.

If you want to get computer friendly then get *Call–A.P.P.L.E.*

Keep up the good work.

Ken Glickfeld

[Editor's note: Glickfeld met our staff at Chicago's Userfest and was so enthusiastic about A.P.P.L.E. he was asked to write down his opinion of our organization and magazine.]

Dear *Call –A.P.P.L.E.,*

Just a quick note to let you know I appreciate your magazine. Concerning the comment in a recent issue about Apple Co. not giving you an advance model of the Macintosh: They missed a fantastic source of advertising and enthusiasm; although your group is unabashedly pro–Apple, users like me depend on your reviews, comments and advice concerning new products because it seems like an honest appraisal.

Keep on writing.

Pat Rise

Satisfied Member

I joined A.P.P.L.E. in January, 1984. Before then I had never touched a "personal" computer and had no experience at all in programming. Having bought a *//*e, I wanted to learn a lot quickly.

I heard about A.P.P.L.E. from a teenager who was in the computer store when I bought the *//*e. He insisted that I join and fortunately I listened to him. At first I was overwhelmed by the technicality of the articles in the magazine; but I thought OK, it is a technical subject and the best way is to read a lot and try things out.

I'd like to advise new members, especially, to make use of the invaluable HOT LINE. I have used this service countless times and still cannot believe that I am getting this kind of personal attention and comprehensive knowledge for the very low price of $25 per year! I seem to get "Butch" Greathouse on the line often. That gentleman has the patience of a saint, the knowledge of a genius and the personality of a crowd pleaser. Thanks Butch.

Geoffrey C. Byard

Letter to Apple

Mr. Steve Jobs
Chairman of the Board
Apple Computer Inc.

Dear Mr. Jobs:

As an Apple II owner since 1978, I have long had an interest in your company, and have been an enthusiastic booster of your products. In learning to use my computer, I have found the Puget Sound users club, A.P.P.L.E., to be of vital importance in understanding its intricacies. Their magazine articles and software have been invaluable. In fact, this organization is one of the chief reasons that I have been anxiously awaiting the coming of your Macintosh. I have assumed that they would do as good a job of supporting it as they have of the Apple II.

Lately, though, there have been indications that they have not been able to get any information on the Mac that is not published in your ads, even though their members are keenly interested in it.

I hope that this is not a permanent policy with you, since I feel that it is to the advantage of every one for there to be a maximum amount of cooperation among all concerned. The members will be more inclined to step up to a Mac, and will in turn urge their friends to do the same.

My own feeling is that without maximum support from the most knowledgeable user's group in the country, I will not get maximum advantage from owning a Mac. And that would indeed be disappointing.

Very Truly,

T. Stewart Harris

A View From "Down Under"

I read with interest, as I always do your magazine, your Editorial in the March issue of *Call–A.P.P.L.E.*

As editor of the Sydney Apple Users Group magazine, I have–for some years–been close to the problems you are outlining, and thus would like to offer my views on at least the Australian scene in the Apple world.

The user community is looked upon, rightly or wrongly, as:

1. Of no value to prospective sales.
2. A deterent to further sales because
   a. The clubs sell their own "cheap stuff".
   b. Copying is rife within the club members.
3. Very useful to send difficult problems to.

What have we done about it? Whilst Apples were imported by an agency, there was not a great deal we

could do. Since the entry of Apple Inc., we have made many personal contacts within the local company and established a good working relationship, deleting their fears about items 1. and 2.

It is worthwile noting that Apple Inc. is pursuing the hobbyist market, unlike most other computer manufacturers. Where there are hobbyists there are "hackers," and there needs to be "hacker" support. We must educate not only the users but also the sellers.

Hans Hoffman
Sydney (Australia) Apple Users Group

An Open Letter to Software Pirates

I am constantly amazed at a practice so prevalent among microcomputer owners that a person who does not participate in it is almost considered a freak. It is commonly referred to as software piracy. I prefer, however, to call it by its correct name – Software THEFT. I am not referring only to copying software for profit or underground distribution, but to any copying of copyrighted software which was not purchased by the user. The law is clear on this point. You have the legal right to copy software which you have purchased for your own use. Any other copying – even if you give, not sell, the copies to friends, school, users' groups, or anyone else – is ILLEGAL. In spite of this, I know people who would not think of violating the 55 mph speed limit, but freely and frequently copy copyrighted software.

I have heard and read more rationalizations for this practice than I could possibly list. One of the most common I hear is "software is overpriced." If you absolutely cannot live without a piece of software, then whatever its price, it should be worth it to you to pay the price. Otherwise, I suggest that you do not buy it, do not copy it, and do not use it. If it truly is overpriced and other people do not buy it or use it, the price will come down or it will disappear. Rather, we should encourage and support such software publishers as A.P.P.L.E., Penguin, and Beagle Bros., who have chosen to reduce their costs and their prices by not copy protecting their software. These publishers sell top-quality software for reasonable prices. I encourage you to buy and use their products, but please do not rip them off by stealing their software.

For those of you whose only ethical standard is "It's OK if I don't get caught," you are right about one thing. If you don't sell bootlegged software you probably won't get caught. To those of you who hold to some form of ethics on morality which says it is wrong to steal, whether or not you might get caught, consider what you are doing when you accept or give copies of copyrighted software and the example you are setting for your children and friends. What you are doing is stealing, regardless of your rationalizations to the contrary.

William G. Wright
4591 Jamboree St.
Oceanside, CA 92056

Modem Problem Solved

I have recently had difficulties with Multi-Tech Modem, Copy Master, and an 80-column card. By a process of elimination, I discovered that the problem was caused by the Disk Controller Card. It passed all program tests. A replacement of the card solved all problems. Without the Modem and Copy card, my Apple II plus worked great with the "bad" card. However, the "bad" Controller card would not let the Modem turn on, since there was a contention for the upper end of RAM with the Copy Master II.

G. Lane Arbuthnot

80-Column Graphics

After installing an Applied Engineering 80-column card in my Apple][+, I found that I could not view any graphics from GBASIC. This was because CP/M had turned on the card, but the desired graphics signal was only available from the 40-column output. At first, I thought it would be a simple matter of issuing a software command to the soft switch on the card to select 40-column video. Unfortunately, I found that CP/M kept the card in the 80-column mode, regardless of what I tried.

I considered using the manual override input on the card. A logic zero forces selection of 40-column video regardless of any software commands. Integrated circuit F14 on the motherboard (pin 4) provides an appropriate signal to drive the override input. For textmode only, this pin is at a logic one, and is at a logic zero for graphics and split screen modes. After making this connection, I found that I could now view graphics. However, in split screen modes, all text was still routed to the 80-column card instead of the text window. If anyone knows how to patch CP/M so that the user can control 40/80 column selection, I'd appreciate finding out how to do it.

The manual that came with the card left a lot to be desired, so I wrote to Applied Engineering about the possibility of obtaining a revised edition. Not only did I receive a revised manual, but a new ROM also. With this ROM, the home command from Applesoft now works properly. With the old ROM, home did not clear the screen.

Mike Powell
6215 146th N.E.
Redmond, WA 98052

ACCESS ALL MICRO
11306 Southland Road
Forest Park, Ohio 45240

TOTAL ACCESS TO
Software
- BUSINESS
- EDUCATIONAL
- GRAPHICS/ UTILITIES
- PERSONAL PLANNING
- WORD PROCESSING
- BOOKS
- ENTERTAINMENT

Hardware and Accessories
- PRINTERS
- DISK DRIVES
- INTERFACES
- MONITORS
- MODEMS
- DISKS
- RIBBONS

CALL
1-800-543-1114
Ohio · Alaska · Hawaii
Call 513-825-5803
CINCINNATI CUSTOMERS USE
OUR SOFTWARE PICK-UP FACILITY

Circle 125 on Reader Service Card

THE MARKETPLACE

The following items are selected by our editorial staff from the large number of product announcements and news releases sent to us daily. They are presented solely to keep you informed about the current market for Apple related hardware and software. As we have not tested many of these products, appearance here does not necessarily represent endorsement or recommendation by A.P.P.L.E. Watch for evaluative reviews in *Call — A.P.P.L.E.*.

FOREIGN LANGUAGE LEARNING SYSTEM

LampLighter Software, Inc. of Melville, New York has anncunced the introduction of "Language Lab," a complete foreign language training system for Apple and other computers.

The "Language Lab" program offers computer-aided Study, Self Quizzing, and interactive Audio Cassette for pronunciation.

Included with the "Language Lab Learning System," is the "Interactive Audio Cassette Adapter." With this feature there is no need to manually locate and play-back pronunciation lessons on a cassette player. "Language Lab" software automatically starts and stops the student's cassette player in coordination with the lessons presented on the computer display.

Another feature of "Language Lab," is its ability to provide performance reviews through computer-generated progress evaluations. These reports are presented as both a performance listing and a progress chart.

The program requires 64K of RAM, one diskette drive, and an audio cassette player. The program may be utilized with either color or monochrome displays.

"Language Lab Learning Systems" are available in Spanish, French, and German, at $235.00 per language.

"Language Lab" is available through *LampLighter Software, Inc., 7 Breton Avenue, Melville, NY 11747, (516) 421-2653.*

SOFTWARE FOR LEARNING FOREIGN LANGUAGES

Sunnyvale, Calif. — Soflight Software has announced a new product line of Language Development Software, which runs on Apple II or Apple *IIe* computers with one disk drive. All programs teach 1000 of the most common words in the target language. When words have more than one meaning, the program allows for these other meanings, along with English translation. A "Teach Yourself Book" is included for additional information in the package that retails for $56.95.

Call Apple—JULY Disk: Apple 31
Title: Market 1
1-4 Oracle 5-8 Helios
File 3878ga03

Currently available languages in the product line include Spanish, French, German, Italian, Biblical Hebrew, Modern Hebrew and Arabic. In the near future Latin, Russian, Polish, Swedish, and Classical Greek will also be available.

Contact: *Howard Piercy, Soflight Software, 2223 Encinal Station, Sunnyvale, CA 94087, (408) 735-0871.*

FRENCH AND SPANISH LEARNING SOFTWARE

Gessler Educational Software has released POKER PARI and ANAGRAMAS HISPANOAMERICANOS. POKER PARI is an all-text card game in French. Players are dealt seven cards, each reviewing one of 14 topics on French culture, grammar or vocabulary. There are hundreds of questions. At the beginning of each hand, a player or team may discard up to two cards if they do not like the topics, and select new ones. The player then chooses either an "easy" or a "hard" question before each play. Hard questions are worth more points. Correct answers are rewarded with "Bravo!." A musical salute encourages the player on after each hand, win or lose.

POKER PARI is an educational game that can be played by individuals, small groups or an entire class for interaction and competition. It is available on disk for the Apple II, II+, and *IIe* with 48K memory, at a suggested retail price of $39.95.

ANAGRAMAS HISPANOAMERICANOS is a geography lesson in Spanish. Its high resolution color graphic depictions of South and Central America, Mexico and the Caribbean are designed for easy teaching and learning. As the names of countries or capitals appear as anagrams on the left hand side of the screen, a colorful map of the entire region appears on the right side, with the area to be identified shown directly above the scrambled letters. If the letters are properly unscrambled, the area returns to its appropriate place on the map to the right; if the letters are not correctly unscrambled, the area remains blank on the map. Correct answers are followed by a "Fabuloso," "Muy Bien," or other encouraging expressions. If an answer is wrong, the correct answer appears on the screen.

This program may be used by people at the beginning, intermediate, or advanced level of Spanish to learn, review and reinforce the names and locations of the countries and capitals of South and Central America, Mexico and the Caribbean.

ANAGRAMAS HISPANOAMERICANOS is available on disk for the Apple II, II+, and *IIe*, with 48K memory, at a suggested retail price of $37.95. Both programs are accompanied by extensive documentation and step-by-step operating manuals.

MEMBER SERVICES ARE FOCUS OF FORTH INTEREST GROUP

San Carlos, CA — The FORTH Interest Group (FIG) is emphasizing member services as one of its prime objectives. FIG is a non-profit, member-supported organization with over 4,800 members and 53 chapters worldwide. Services offered include publication of FORTH Dimensions magazine (for members), the FIG-Tree, an online computer data base, various FORTH manuals and other publications, the Job Registry, various conferences, the annual FORTH Convention (November 16-17, 1984 in Palo Alto, CA), a user library, membership directory, speaker bureau, catalog of FORTH software, educational tapes, and more.

For additional information on FIG membership ($15.00/yr, $27.00/yr foreign) call the *FIG HOT LINE (415) 962-8653* or write to *FIG, P.O. Box 1105, San Carlos, CA 94070.*

APPLE II ON THE GO

The new Computer Cart is now available for an introductory price of $54.95, plus $3.50 shipping and handling.

The Computer Card fits Apple II, II+, and IIe with two disk drives, and allows you to carry your Apple from office to home, to club meetings, school, repair shop, or anywhere in a fast, safe, and efficient manner.

In just a few minutes, you can unplug your video and power cables, put disk drives in position, zip up, and be on your way. You do not have to disconnect your disk drive cables to use the Computer Cart.

There is room available in the oversize pocket for 1 or 2 boxes of diskettes, and for your most popular documentation manuals and power cords as well. Everything you need is in one place. An internal pocket holds joysticks or accessories up to 3" wide x 8" long.

The Computer Cart is made of strong 1000 denier Dupont Cordura nylon with a waterproof urethane coating. Three 1" nylon straps with 800 lb. tensile strength fully encircle all components to prevent slippage when carrying your valuable equipment. (A 2" wide shoulder strap is optional for $4.95.) The Computer Cart rolls up for storage when not in use.

Contact: *Bob Preston, Technology One, 107 Delaware Avenue, Albany, NY 12202, (518) 436#9933.*

AN INTERACTIVE INFORMATION MANAGEMENT SYSTEM FOR THE MACINTOSH

San Francisco, CA — Odesta has introduced a product, Odesta Helix, for the Apple Macintosh computer. Odesta Helix is an interactive information management system designed to take special advantage of the Macintosh windows, icons, pull-down windows and mouse. The program makes possible a flexible, "give and take" interaction with the Macintosh for information and knowledge-based activities such as free-form data base management, text processing, statistical analysis, and a wide range of form-based input, query and reporting.

Distinguishing Odesta Helix from previous data base and information management systems is the fact that it doesn't limit or restrict the person using the program. For instance, other programs require an understanding of a programming language, the extensive use of command keys, and dictate the size of documents and the form of entries and searches. Odesta Helix, on the other hand, places no restrictions or limitations on length, content or associations.

The retail price for Odesta Helix is $395.00.

Contact: *Odesta Corporation, 3186 Doolittle Drive, Northbrook, IL 60062, (312) 498-5615.*

SOFTWARE ADVENTURERS.

Activision wants you to venture where no software designer has gone before. Our new move into the home computer software market can unleash your creativity.

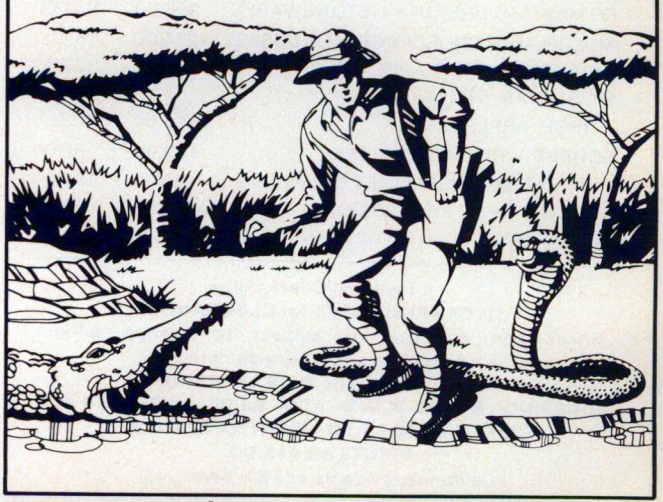
If you're an expert in Assembly language programming on an Apple II or Commodore 64, and have the discs to prove it, now is the time to join Activision. We're pushing to the limits on entertainment and learning software through superior graphics and captivating storylines. The next step is up to you.

Activision offers you the software adventure of a lifetime through our "no boundaries" environment for top-notch software developers. The next generation of home computer software will be written by people of vision, who are able to turn that vision into reality.

In addition to standard employment, we offer open work arrangements and contract possibilities, in flexible geographic locations. Both original and conversion responsibilities are available.

For more information about the incredible adventure that awaits you, please call Barbara Thompson at (415) 960-0410, ext. 5213, or write to her in care of Activision, Inc., P.O. Drawer 7288, Mountain View, CA 94039. We are an equal opportunity employer.

ACTIVISION®



Circle 61 on Reader Service Card

Laing Electronics

10382 STANFORD AVE.
SUITE E-203
GARDEN GROVE, CA 92640
PHONE: (714) 534-4216

MICROSOFT 16K CARD
$59.00

★ HARDWARE FOR THE APPLE II ★

	LIST	YOUR COST
HAYES MICROMODEM II	379.00	258.00
HAYES SMARTMODEM 300 BAUD	289.00	199.00
HAYES SMARTMODEM 1200 BAUD	699.00	498.00
NOVATION APPLE CAT II	389.00	268.00
MICROSOFT Z-80 SOFTCARD CP/M	345.00	219.00
MICROSOFT PREMIUM PACKAGE (Z-80, VIDEX 80 COL., 16K CARD)	695.00	490.00
PCPI 6 Mhz APPLI-CARD (Z-80 CP/M)	375.00	279.00
VIDEX VIDEOTERM 80 COLUMN CARD (SOFTSWITCH, AND INV. CHAR. SET)	319.00	225.00
WIZARD 80 COLUMN CARD	249.00	149.00
WIZARD BUFFERED PINTER INTERFACE (16K BUFF + GRAPHICS)	179.00	129.00
(32K BUFF + GRAPHICS)	219.00	153.00
NEC HI-RES GREEN MONITOR (JB1201)	285.00	153.00
MICRO-SCI A2 APPLE COMPATIABLE DISK DRIVE W/O CONTROLLER	345.00	229.00
MICRO-SCI A2 DRIVE WITH CONTROLLER	445.00	299.00
OKIDATA MICROLINE 92/P PRINTER	699.00	498.00
PROWRITER 8510AP PRINTER	495.00	349.00
KENSINGTON SYSTEM SAVER FAN	89.95	65.00
GIBSON LIGHT PEN/LPS II (PENTRACK)	349.00	239.00
VISTA V1200 CART. DISK SYSTEM 6.2MB	1840.00	999.00
VISTA A-800 8" DISK CONTROLLER	379.00	249.00
VISTA SOLO 5¼" DISK DRIVE	299.00	195.00
8" DRIVE SUBSYSTEM WITH CONTROLLER 2.4MB	1900.00	995.00

★ VERBATIM DISKS ★

100 DATALIFE 5¼" (MD525-01-18158)	490.00	239.00
50 DATALIFE 5¼" (MD525-01-18158)	245.00	124.00

★ SOFTWARE FOR THE APPLE II ★

PFS: FILE (SOFTWARE PUB.)	125.00	76.00
PFS: REPORT (SOFTWARE PUB.)	125.00	76.00
PFS: GRAPH (SOFTWARE PUB.)	125.00	76.00
DB MASTER REG. VERSION (STONEWARE)	229.00	152.00
DB MASTER VERSION #4 (STONEWARE)	350.00	231.00
MULTIPLAN (MICROSOFT) CP/M OR DOS3.3	275.00	167.00
WORDSTAR (MICRO-PRO CP/M)	395.00	327.00
WORDSTAR WITH Z-CARD (PCPI APPLI-CARD)	495.00	425.00
SCREEN WRITER II (ON-LINE)	129.95	86.00
FORTRAN-80 (MICROSOFT CP/M)	195.00	129.00
ASCII EXPRESS PROFESSIONAL	129.95	85.00
Z-TERM PRO. (SOUTHWESTERN DATA)	149.95	98.00

If Personal Check, Allow
10 BANKING DAYS for CLEARANCE.
Prices and specifications subject to change without notice. Unless otherwise noted,
ALL PRICES are PLUS SHIPPING.
SHIPPING & HANDLING—ADD $5.00 UPS SURFACE
(DISK) DRIVES $7.50—MONITORS $10.00—
PRINTERS $15.00.
California residents add 6% Sales Tax.

Circle 124 on Reader Service Card

NEW VERSION OF MEGAWRITER FOR APPLE II & *IIe*

San Diego — Megahaus Corporation has released an updated version of MegaWriter for the Apple II and *IIe* computers. Megawriter is a full-featured word processor with mail merge that retails for $99.95.

The new program comes complete on a single disk. Improvements include a faster boot time, settable tabs, and the ability to read and write both Apple DOS files (such as VisiCalc print files) and Pascal files.

MegaWriter automatically configures itself to either the Apple II or *IIe* computers and takes full advantage of the *IIe* keyboard. MegaWriter allows extensive print formatting and is easy to use.

The MegaWriter family of programs include MegaSpell (a spelling checker) and MegaFinder (a file manager and report generator).

Contact: *Paul La Costa, Megahaus Corporation, 5703 Oberline Drive, San Diego, CA 92121, (619) 450-1230.*

MEGAMERGE FOR THE MACINTOSH

San Diego, CA — Megahaus Corporation began shipping MegaMerge, a mail merging program that turns MacWrite into a powerful home and business word processor. MegaMerge sells for $125.

MegaMerge can produce hundreds of form letters, mailing lists, labels, etc., from just one "master" document and a list. MegaMerge also provides "boiler plating" capabilities, so that you can merge large paragraphs of information into your printed documents.

MegaMerge can also "chain" documents together, so that you can print several documents as thought they were one.

Contact: *Paul La Costa, Megahaus Corporation, 5703 Oberline Drive, San Diego, CA 92121, (619) 450-1230.*

APPLE LOGO *II*

Montreal — Logo Computer Systems, Inc. of Lachine, Quebec announced the release of Apple Logo *II* developed especially for the new Apple *IIc* microcomputer.

Distributed by Apple Computer, Inc. at a retail price of $100 (U.S.), Apple Logo *II* is designed to take advantage of the special features and larger memory of the Apple *IIe* and Apple *IIc* personal computers.

Because Apple Logo *II* takes advantage of 128K of memory, it has nearly three times the memory available to store, retrieve and edit programs than the original Apple Logo. Because Apple Logo *II* works with ProDOS, Apple's high-level operating system, data can be stored on a standard floppy disk or on a ProFile hard disk.

Apple Logo *II* comes with special help messages, colorful sample programs, and a comprehensive tutorial disk — Apple Presents Apple Logo *II* — that walks you smoothly from beginning turtle graphics to an understanding of Logo's many powerful programming concepts such as recursion and the manipulation of words and lists.

Contact: *Douglas R. Long, Director of Communications, (514) 631-7081.*

APPLE INTRODUCES NEW PROGRAM FOR MACINTOSH SOFTWARE DEVELOPERS

Cupertino, CA — Apple Computer has announced a new program designed to help independent Macintosh software and hardware developers get their products to market quickly. Called MacCollege, the program assists developers who are in the final stages of product development by offering three-day sessions consisting of technical information, private consultation, and laboratory time.

Classes will be held at the new Macintosh Software Mill, a dedicated facility located at the company's headquarters in Cupertino. Open 24-hours a day to programmers enrolled in MacCollege, the Software Mill houses a lecture hall and private cubicles and equipment for programming.

MacCollege is designed for experienced programmers who are familiar with the Macintosh development environment and who are certified under Apple's Certified/Registered Developers program, a corporate support program for developers of Apple-compatible products. The curriculum covers such subjects as user interface, program design, and real time de-bugging.

Sessions are scheduled from June 11, 1984 until mid-November and will be taught by members of the Macintosh Developers Group and Macintosh software engineers. For registration information contact: *Macintosh Software Mill* at *(408) 973-4387.*

MCPIC! — READY-TO-USE PICTURES FOR MACINTOSH

The new McPic! program was designed to let those with little or no drawing ability make full use of the Macintosh's graphic capabilities. This push-button picture library gives the user 130 pictures created by professional artists, in a variety of art styles.

The McPictures can be used immediately with the Click of a Mouse, or, they can be customized with MacPaint/MacWrite. The artwork can be reduced or expanded in size—the tones and patterns can be changed—they can be duplicated, modified, changed, cloned or combined with other pictures, or a picture segment can be selected and used by itself. Words can also be added.

The 130 McPictures cover a wide range of catagories and provide a graphic solution to almost any situation for business, home, school, organizational or recreational use.

McPic! is $49.95 with disk and a Hint Manual which provides ideas for making the most of this push-button picture library. *Magnum Software, 21115 Devonshire St., Ste. 337, Chatsworth, CA 91311, (818) 700-0510.*

FREE LISA/MACINTOSH PUBLICATION

Aptos, CA — A free newsletter for users of Apple's Lisa and Macintosh computers has just completed its first year of publication. The monthly publication is named Semaphore Signal, and is unique in that it is created, edited and printed using only a Lisa. Macintosh coverage in Signal began with the February 1984 issue. Subscriptions to Semaphore Signal are free to any user submitting the serial number of the Lisa or Macintosh they use to: *Semaphore Signal, 207 Granada Drive, Aptos, CA 95003.*

Other readers may subscribe to Signal for $10 per ten issues in North America, $20 outside of North America.

VIDEO ACQUISITION SYSTEM FOR APPLE II

Needham, MA — DIGITAL VISION, INC., announces the introduction of COMPUTEREYES™, an ultra-low-cost video acquisition system for the Apple II series. Priced at $129.95, COMPUTEREYES represents a cost-effective means of capturing real-world images on the Apple's hi-res display. For $349.95, a complete system including COMPUTEREYES and a high-quality video camera is available.

COMPUTEREYES is an innovative slow-scan device that connects between any standard video source (video tape recorder, video camera, videodisk, etc.) and the Apple's game I/O socket. It therefore uses no peripheral slots. Under software control, a b/w image is acquired in less than five seconds. A multi-scan mode also provides realistic grey-scale images.

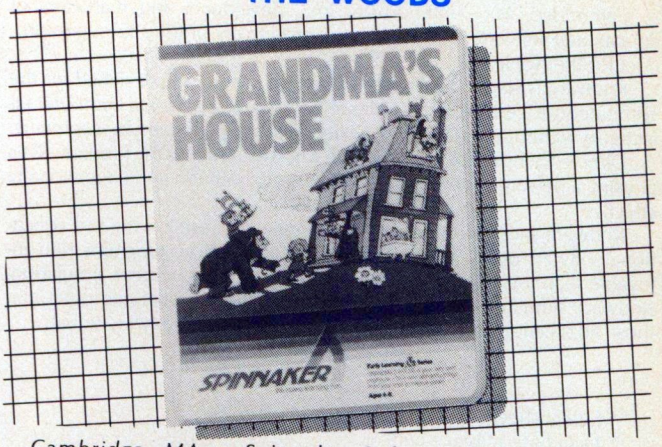
Applications of COMPUTEREYES include pattern recognition, security, quality control, spatial measurement, robotics and artificial intelligence, industrial controls, computer art, education, and entertainment.

Software provided with the system includes: machine language image capture routines; a menu-driven executive that provides everything even first-time users need to capture images; image save-to-disk capability; image packing and unpacking routines that save disk space and speed loading and saving; and a tutorial as an aid in developing applications.

The COMPUTEREYES package includes interface module, cable, software support on disk, owner's manual, and one year warranty. The system is currently available for 48K Apple II series and compatibles, with Applesoft and DOS 3.3. A demo disk that simulates and describes the operation of COMPUTEREYES is available for $10.00.

For further information, contact: *DIGITAL VISION, INC., 14 Oak Street, Suite 2, Needham, MA 02192 (617) 444-9040.*

OVER THE RIVER AND THROUGH THE WOODS



Cambridge, MA — Spinnaker Software Corp. introduces GRANDMA'S HOUSE™ for the new Apple *IIc* computer.

What could be more fun than a visit to GRANDMA'S HOUSE? It's a game that lets children create their own perfect playhouse by furnishing it with wonderful things from different magical places. By using the mouse, players can quickly move from one fanciful place to another, picking up new decorations for this stately Victorian manor.

GRANDMA'S HOUSE, developed for Spinnaker Software Corp. by Interactive Picture Systems, helps children learn to design and create within a unique format. Children choose who they want to be from a cast of thirty characters. Then off they go to GRANDMA'S HOUSE, the starting point for their explorations of special places. Once treasures are collected, they are brought back to GRANDMA'S HOUSE to be arranged. It is fun to figure out where to put the tiger that was brought from the jungle; the parking meter from the city; or the bathtub from the furniture store.

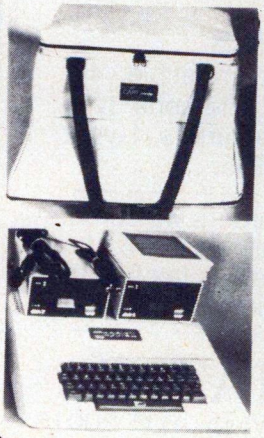
"Your Apple* II is now Portable"

© 1984 Technology One

don't risk carrying it any other way!

- *fits Apple* II, II + and II E with two disc drives*
- *1000 Denier Cordura nylon construction (silver only)*
- *1 year warranty*
- *pockets to carry power cords, 10 or 20 diskettes, software manuals, joysticks, etc.*
- *internal strap holds disc drives securely*
- *no need to disconnect disc drive cables*
- *leather hand grip standard*
- *optional shoulder strap — $4.95*

(Franklin Available July '84)



. . . now make it easy to protect your investment
— dealer inquiries invited —
*Apple Computer Company

COMPUTER CART™

$54.95 plus 3.50 shipping & handling

Order from:
TECHNOLOGY ONE
P.O. Box 1497 (mail orders only)
Albany, New York 12201
(518) 436-9933 (dealer inquiries only)

Name _____
Address _____
City _____ State _____ Zip _____
☐ VISA ☐ MasterCard ☐ Check or M.O.
Exp. date _____
Card # _____
N.Y. Residents add 7% tax

Circle 123 on Reader Service Card

KNOWARE PRODUCT FOR APPLE *IIc*

Cambridge, MA, — Knoware, Inc., developers and marketers of educational software, announced the availability of their Knoware product for the Apple *IIc*.

Knoware is starter applications software specifically designed for the computer novice who wants to learn how to use a personal computer effectively without the stress and anxiety often experienced by beginners.

Directed at business professionals, Knoware teaches the fundamentals of using a personal computer in a way that is both fun and educationally effective.

Knoware succeeds at teaching beginners because it is the first educational software product to integrate starter applications programs with enjoyable learning. By taking you through a business career simulation, Knoware simultaneously introduces you to the personal computer and those applications which are most frequently used today including:

* Spreadsheet
* Data base management
* Text editing
* Financial decision-making
* Simple graphics

Knoware requires no prior experience, no manuals, no classes, and very little time or money. An easy-to-learn keyboard tutorial makes getting started easy. As a result, within two hours, someone using Knoware for the first time will become confident and comfortable about using personal computers.

To begin the Knoware experience, users follow a career path from mail clerk to chairman of the board, and in doing so, learn the most important applications for personal computing.

Contact: *Nancy C. Mains, Knoware, Inc., 301 Vassar Street, Cambridge, MA 02139, (617) 576-3821.*

Highest Quality, **Lifetime Guarantee!**

DISKETTES

$1.65 5¼" soft-sectored, hub ring, envelopes, double density, double-sided on APPLE drives -- 100 for $155, 100 single-sided for $149.

Hard plastic stand-up 10-diskette carrying cases $2.75 each, 4 for $10 (beige, black, blue, green, grey, red, yellow). Smoked-plastic flip-top 75 diskette file cases, $19.50. Heavy-duty nibbling tool, $22.

Disk Drives

$199 100% APPLE-compatible, 40-track, full-size, Siemens type quality drives, with manufacturer's 1-year warrantee. Controller card, $65.

COD & VISA/Master Card orders welcome. Add $4 for shipping & handling (only $2 for orders under $50) plus 6% sales tax for DC residents. Send for our catalog.

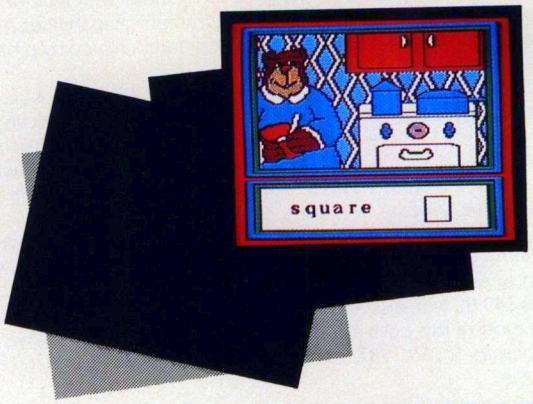
VF ASSOCIATES

6327 Western Ave., N.W., Wash., D.C. 20015

(202) 363-1313

Circle 115 on Reader Service Card

STICKYBEAR SHAPES ENHANCED FOR APPLE *IIc*



Weekly Reader Family Software, a division of Xerox Education Publications, has announced the enhancement of its early learning software program, Stickybear Shapes, for use on the Apple *IIc*.

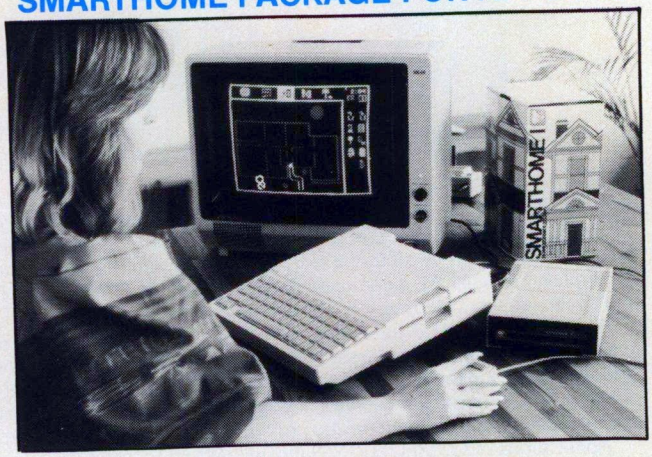
A colorful introduction to shapes for children 3 to 6 years old has been adapted so that, in addition to operating on Apple *IIc*'s keyboard, the program also accepts input from the AppleMouse *IIc* — the hand-held pointing device that virtually eliminates the need for complicated keyboard commands.

Stickybear Shapes teaches young children the concept of shapes through colorfully animated activities, including "Name a Shape," "Pick a Shape" and "Find a Shape." When a child is able to complete the request, he or she is rewarded with happy, colorful animation and music, notes its creator, Richard Hefter.

Stickybear Shapes, and the complete line of Weekly Reader Family Software's early learning software packages, come with a 32-page hardcover book, a full color poster, a user's guide and colorful stickers to augment the child's learning experience.

For further information contact: *Fritz Luecke or Judy Carlson, Xerox Education Publications, 245 Long Hill Road, Middletown, CT 06475, (203) 347-7251.*

SMARTHOME PACKAGE FOR APPLE *IIc*



Boulder, CO — CyberLYNX Computer Products, Inc. announced a version of its SMARTHOME peripheral and software package which takes advantage of certain features of the new Apple® *IIc* home computer.

SMARTHOME, a completely icon graphics based program, allows the new Apple computer to automate the home; to control lights and appliances throughout the house and monitor a security system. It is programmed entirely with the Apple *IIc* mouse and uses enhanced icon graphic features. The Apple *IIc* programmable serial ports allow interface to SMARTHOME without additional hardware. It connects using the same cable as the new SCRIBE printer for the Apple *IIc*.

The SMARTHOME is entirely wireless and user installable and the "starter package" will sell for $499.00.

Contact: *CyberLynx, Cinda Johnson, 4828 Sterling Dr., Boulder, CO 80301, (303) 444-7733.*

Unclassifieds

Members Non-commercial rate .05/word
Members commercial rate .25/word
All others: 1.00/word, $25 minimum
Send ad copy with payment to editorial office. Unclassified copy must be received by the first day of the month preceding the date of issue.

RENT SOFTWARE—$5 monthly. No membership fee. Send large SASE to: Preowned Software, 4229 Baker Ave NW, Seattle, WA 98107 for list.

STOCK MARKET SOFTWARE WANTED—Will pay for programs that use the RTR data disks. Especially wanted are USER routines for the DJMA (Apple II version), i.e.,Wells Wilder, %A, Magic T, Cycle Analysis, etc. Send information to W.Wolfe, Box 238, New York, NY 10470.

NOTCH-0 PUNCH—Want to write on the back of your disk? Use our self-aligning Notch-o punch to add a write-protect notch. $8.95 + $1.05 shipping (CA res. add sales tax). Geisler Enterprises, 20333 Old Town Rd., Star Rte. -2, Tehachapi, CA 93561.

AIR NAV WORKSHOP (II +, //e), fully-customizable, navigate any area, adjustable winds, automatic flight path replay, much more,$40.Check, MC/V, UPS-COD. SPACE-TIME ASSOCIATES, 2039-CA Country Club Drive, Manchester, NH 03102.(603)625-1094. Free info.

INTERFACE CARDS FOR WHOLESALE PRICES—16K RAM Card $49, Z80 Card $39, 128K RAM Card $149, 8088 Card $199, 6809 Card $139, 80 Column Card $59, and 30 more different kinds of cards for special prices. HUNTER ELECTRONIC CO., P.O. BOX 53-130, TAIPEI,TAIWAN, REP. OF CHINA.

APPLE CARDS AT LOWEST PRICES!! Everything you need for your Apple II computer at prices you never saw anywhere before. Buy direct from us in the Far East and save money on everything!!

RAM Card 16k memory	$46.00
RAM Card 128k memory	$189.00
80-Column Card	$86.00
Z-80 CP/M Card	$52.00
EPROM Writer Card	$79.50
Remote Automatic Mouth	$36.00
Serial Interface Card	$49.50
Disc Drive Controller Card	$55.00
Disc Copy Card	$45.00
Self-centering Joystick	$17.50

All Payments by cashiers check only, sorry no personal checks. Send your name and address for our complete catalog. Or leave Sourcemail at BBV970. COMMUNICATIONS ASSOCIATES, GPO BOX 7779, HONG KONG.

ALMOST WHOLESALE—Save BIG BUCKS on almost all Hardware & Software needs. Usually 20% to 45% below retail. Monthly specials. Send self-addressed stamped envelope for latest price list, or contact us to request quotes on specific items. COMPUTER PRODUCTS UNLIMITED, Apple Dept., 11116 East 46th Ave, CompuServe (73075,613). SOURCE (TCG249).

ARCADERS DO NOT READ—Hegira-Tale of a Galactic Exile, was developed by a professional programmer of large computers, who has also published science fiction stories. It is fun to play. So if you're tired of arcade games,try an S.F. adventure game. Send $15 to Ted Butler 12545 SE 270th St. Kent WA 98031. Wash. res. add 7.6% sales tax. Hegira requires a 48k Apple II, II + *1, or* //e, with DOS 3.3.

DISKETTES!!! Maxell MD-1, SS/DD, $25.00/10. Generic SS/DD $17.00/10. Add $1.00 Shipping (IL residents add 6% sales tax). VISA/MC. Cottage Resources, Suite 243, 1032 Stevenson Drive, Springfield, IL 62703.

DATABASE GENERATOR—Create your own customized, self-booting, independent databases without programming. Each holds up to 1150 records. Great for lists, labels, and files of all sorts. Both generator and databases operate via user-friendly menus. Requires II +, //e, //c with Apple Pascal, 80 column card and two drives. Send for free information to Alyeska Software, 3911 Rose, Seal Beach, CA 90740.

DISK DRIVES, MCI, SS/DD, BRAND NEW, $200.00 EACH, CALL QUICK! (206) 392-8433.

THE DUPAGE AMATEUR CLUB's Ham/Computer fest will be held on Sunday July 8, 1984 at the Downers Grove, Illinois American Legion Post 80. It will feature a large outdoor flea market (swapper's row). Indoor commercial exhibits available. General admission $2.00, $3.00 for swapper's sellers. Tickets available at the gate ONLY. Plenty of free parking. Good food and drink available. Ham Radio Talk-in 145.490 Mhz Rpt or 223.74 Mhz Rpt. For more information send SASE to W9DUP, P.O. Box 71 Clarendon Hills, IL 60514 or call (312) 971-3298 between 8 a.m. and 9 p.m., everyday BUT Wednesday and Sunday (thank you).

OVER 2000 SHAPES—ready-to-use tables on 2 disks. Character sets, game & animation sets, spaceships, animals, flowers, ghosts, beasts, trees, houses, snowflakes, flowchart symbols & more, in various sizes. $29.50 includes instructions, viewing & demo programs. 48K Apple II's. LUSTER SOFTWARE (Dept. CA), 541 G Street, Salida, CO 81201.

APPLE SILENTYPE PRINTER and software for true descenders, paging and double size graphics. Good student printer. Best offer over $150. Please call David at (503) 754-6363.

DISKS $1.95 EACH—NASHUA SS/SD 5¼__ soft sectored disks. Order in quantities of ten. No fancy boxes, just HIGH QUALITY, low cost disks. Excellent for Apple & TRS-80 models. Sigma Software, P.O. Box 241, Simpsonville, MD 21150. Add $2 for shipping.

THE COMPLETE PAPER ROUTE ACCOUNTANT II, $35. Complete control over your small business. Print route lists, enter payments, print bills as well as current balance sheets for every customer. This program a MUST for larger routes. Written by a former carrier. Requires Apple or "look-alike" min. 48K. Sigma Software, P.O. Box 241, Simpsonville, MD 21150. Add $2 for shipping.

PRINTER SETUP PROGRAM $25. Menu driven setup for EPSON or "look-alike" for any fonts available on your printer. Includes Double-Strike, Double-Size, Emphasized, Condensed and 17 others. Works with almost any word processor from APPLE WRITER to WORDSTAR. Requires Apple or "look-alike" with min. 48K. Sigma Software, P.O. Box 241, Simpsonville, MD 21150. Add $2 for shipping.

DISK O' UTILITIES! Thirteen useful Apple utilities on disk for about a buck each! A Bargain! Send $13.95 to: BROADWAY SOFTWARE, Suite 136, 660 Amsterdam, New York City 10025 (VISA/MC).

SKYFORTH: COMPILED LANGUAGE SYSTEM—super fast, easy to use full feature programming for the Apple II family. Includes short and long integers and floating point numbers. Advanced graphics with vector, block and character shapes and color fill. Print formatting, fast disk access, sorting and more. Reviewed in Softalk 10/83. $95 (WA add 7.9%). Call or write for more information. TOSCH Information Management, 16025 10th Ave SW, Seattle, WA 98166 (206) 246-3839.

IDS 440G PRINTER WITH INTERFACE CARD, $200/offer. Versa-writer 'like' digitizer, $70/offer. Rev 7 keyboard plus Videx Keyboard Enhancer II, $115. Tom Tosch (206) 246-3839. 16025 10th Ave. SW, Seattle, WA 98166.

LOW SOFTWARE PRICES! Multiplan—$136.50. Check out our other fantastic prices. Write for our FREE price list! KERR SOFTWARE, Box 5301-A Long Beach, CA 90805. Or call (213) 248-8193.

GIANT SLOTHS HATE RICE. Who cares? All you're really interested in is winning at Ultima II and III , right? World's easiest to use character editors. $9 each or $14 for both disks. Michael Scanlin, 34 Giralda, Long Beach, CA 90803. Specify II or III.

Print EXACTLY what is on your Apple's screen
WITHOUT affecting whatever program is running

SSD™ SUPER SCREEN DUMP

DO NOT EXIT YOUR PROGRAM TO DUMP THE SCREEN, SIMPLY PRESS THE BUTTON (ON THE INCLUDED HARDWARE) AND TYPE A ONE LETTER COMMAND, *ALLOWING YOU TO PRINT*:

**** ALL TEXT (from page 1 or 2); 40 columns, 80 columns (with //e 80 column board)
**** ALL GRAPHICS (from page 1 or 2); LOW RES, HI RES
**** MIXED GRAPHICS AND TEXT
**** TITLES AND COMMENTS with your printout
**** KOALA SCREEN (normal or magnified)

SYSTEM REQUIREMENTS

- APPLE//e or APPLE][+ with 48K (or compatible)
- For all SSD features:
  APPLE (IMAGEWRITER or DMP) or EPSON/GEMINI or compatible
- For all SSD features except low res graphics:
  Graphics printer interface (like GRAPPLER) with any printer supported by that interface
- For all of SSD text features (no graphics):
  Any printer with any printer interface
- Use with any unprotected DOS 3.3 software loaded below $9000 (36864)

SSD KIT $41

KIT Contains:
USER MANUAL
COPYABLE DISC (with SSD Software)
SMALL BOARD (insert in any slot)

ProDOS and other versions available soon!

*** PRINT WHILE USING DOS OR BASIC COMMANDS
*** CHANGE PRINTER SET UP WHILE ANOTHER PROGRAM IS RUNNING

SCHWARTZ COMPUTER CONSULTING — MYCROFT-SOFT SOFTWARE
3868 Cedar Hill Road, Victoria, B.C., CANADA, V8P 3Z6
Please send me ☐SSD KIT $41 ☐MANUAL ONLY $1 ☐MORE INFO
Enclosed is my ☐CHEQUE ☐MONEY ORDER FOR $_____
Name_____
Address_____
City/State/Zip_____

APPLE TM Apple Computer Inc EPSON TM Epson Corp GEMINI TM Star Micronics
KOALA TM Koala Tech Corp GRAPPLER TM Orange Micro

Circle 126 on Reader Service Card

NOW YOUR APPLE® CAN HOOK TO ANY MAINFRAME.

Apple is a registered trademark of Apple Computer, Inc., CP/M is a registered trademark of Digital Research, Inc., Dow Jones News/Retrieval is a registered trademark of Dow Jones, Inc. The Source is a service mark of Source Telecomputing Corporation, CompuServe is a registered trademark of CompuServe, Inc.

If you have an Apple II, Apple II Plus, or Apple IIe, we have good news for you. Now there are two inexpensive software programs that can turn your Apple into a much more valuable tool.

Softerm 1 connects you with information services.
Softerm 1 lets you retrieve information from services such as *The Source,*℠ *CompuServe®,* and *Dow Jones News/Retrieval®.* Plus, gives you the ability to access bulletin boards and send or receive electronic mail. Other features include user-defined keyboard macros, built-in phone book for automatic dialing, terminal mode line capture simultaneously to print or disk, copy screen to print or disk, and terminal status display.

Softerm 2 connects you with your company's computer.
This expanded version of Softerm lets you gain access to the information stored in your company's main computer from your home or office. With either version of Softerm, you can down load information into your Apple and capture it on your own disk in any format you choose—DOS, CP/M®, or Pascal. Also included with Softerm is a source program for your host computer to ensure compatibility with Softerm's file transfer capabilities.

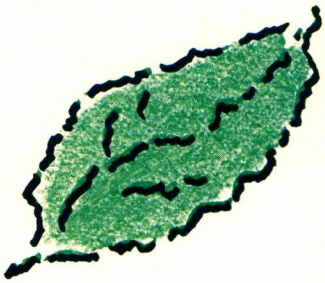
Makes your Apple work exactly like any major terminal.
Softerm 2 provides complete emulation of these terminals: ADDS Regent 20, 25, 40, 60; ADDS Viewpoint; Data General D200; Datapoint 3601; DEC VT102, VT52; Hazeltine 1400, 1410, 1500, 1520; Hewlett-Packard 2622A; Honeywell VIP7205; VIP7801, VIP7803; IBM 3101 Model 10 and 20; Lear Siegler ADM-3A, ADM-5 and TeleVideo 910, 925, 950. And the list is growing all the time. We'll send you a User's Guide, handy reference card, and a telephone number to call if you need more assistance.

Softerm can't become obsolete.
We constantly improve Softerm. And we make those improvements available to you 24 hours a day, 7 days a week. To update your program, just dial our computer and transfer the improvements to your disk.

Get Softerm now.
It's the inexpensive way to turn your Apple into a much more valuable tool for your home or office. Softerm 1 is only $135; Softerm 2 is $195. Both are available now from your local dealer or from Softronics, Inc. For more information or to order, call (901) 683-6850.

SOFTRONICS
3639 New Getwell Road, Suite 10
Memphis, TN 38118. (901) 683-6850

Circle 113 on Reader Service Card



Only once before was an apple made this tempting.

Habadex for Macintosh.™ The ultimate desktop management software. It dials your phone. Sorts hundreds of records. Prints flexible reports. Produces form letters and mailing labels. And it's now running at Apple dealers everywhere.

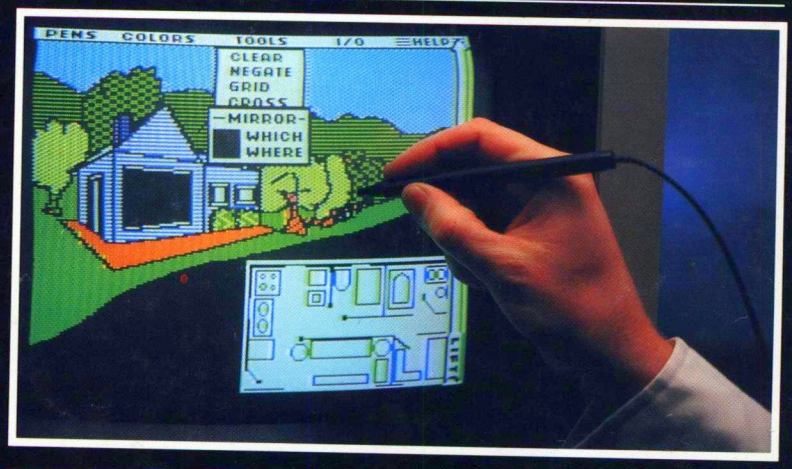
hsi **habadex**™

Making Apples more tempting than ever.

Habadex is a trademark of Haba Systems, Inc. 15154 Stagg Street, Van Nuys, California 91405, (818) 901-8828. Macintosh is a trademark of Apple Computer, Inc.

Circle 64 on Reader Service Card

Now...Draw On Your Imagination



Introducing The Gibson Light Pen System.™

The link between mind and machine has arrived. Suddenly you're free...free to translate your every thought into professional quality computer graphics... just by touching your screen.

The Gibson Light Pen System software features *icon* menus that offer easy access to powerful graphics tools such as symbol libraries, geometric shapes, mirror-imaging, magnification and complete color and pattern editing. Even if you're not a graphic artist, you can design, diagram and draw with precision at high-speed, in high-resolution, and in full-color...right on your screen.

COMPLETE WITH FIVE SOFTWARE SYSTEMS TO MAXIMIZE YOUR CREATIVE OPTIONS.
The Gibson Light Pen System comes complete with all you need to draw, paint, design, score music and learn animation.

DRAW FREEHAND WITH PENPAINTER.™
A full range of drawing tools, shapes, patterns and colors to draw or paint virtually anything on your screen.

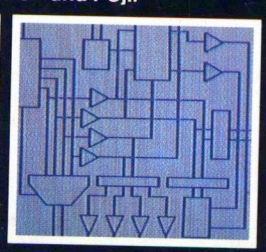
DESIGN PRECISION DIAGRAMS WITH PENDESIGNER.™
Turn your computer into your own graphic design studio. A complete selection of templates make perfect business and architectural diagrams, technical drawings and engineering schematics a snap.

CREATE COMPUTERIZED ANIMATION WITH PENANIMATOR.™
All that you need to learn the basics of animation. Develop your own animation sequences, and bring your screen to life.

COMPOSE MUSIC WITH PENMUSICIAN.™
Score computerized melodies with incredible ease at the touch of your pen.

CREATE YOUR OWN LIGHT PEN APPLICATIONS WITH THE PENTRAK LANGUAGE SYSTEM.™
Take advantage of the software features, and customize your own light pen programs.

NOW AVAILABLE FOR THE APPLE II® SERIES
Coming soon for the IBM PC™ and PCjr.™



Koala Technologies™
800-KOA-BEAR

The Gibson Light Pen System

Touch the Magic of Light

Circle 65 on Reader Service Card